DOCUMENT RESUME

ED 082 491 EM 011 460

AUTHOR	Beyer, William Fred, III
TITLE	A Graphics System for Pole-Zero Map Analysis.
INSTITUTION	North Carolina Univ., Chapel Hill. Dept. of Computer Science.
PUB DATE	70
NOTE	104p.; Thesis submitted to the Department of Computer and Information Science of the University of North Carolina
EDRS PRICE	MF-$0.65 HC-$6.58
DESCRIPTORS	*Computer Assisted Instruction; *Computer Graphics; Computer Science; *Computer Science Education; *Interaction; *Man Machine Systems; Masters Theses; Technical Reports
IDENTIFIERS	Automatic Control Systems; IBM Model 1 Display Unit; IBM System 360 Model 40 Computer; Pole Zero Map Analysis

ABSTRACT

Computer scientists have developed an interactive, graphical display system for pole-zero map analysis. They designed it for use as an educational tool in teaching introductory courses in automatic control systems. The facilities allow the user to specify a control system and an input function in the form of a pole-zero map and then examine the resulting transient response. The system's hardware consists of an IBM Model 1 Display Unit attached to an IBM System/360 Model 40 computer. (Author)

FILMED FROM BEST AVAILABLE COPY

ED 082491

University of North Carolina at Chapel Hill

Department of Computer and Information Science

EM 011 460

ED 082491

A GRAPHICS SYSTEM FOR POLE-ZERO MAP ANALYSIS

by

William Fred Beyer, III

A thesis submitted to the faculty of the University of North Carolina at Chapel Hill in partial fulfillment of the requirements of the degree of Master of Science in the Department of Computer and Information Science

Chapel Hill

1970

U.S. DEPARTMENT OF HEALTH, EDUCATION & WELFARE
NATIONAL INSTITUTE OF EDUCATION
THIS DOCUMENT HAS BEEN REPRODUCED EXACTLY AS RECEIVED FROM THE PERSON OR ORGANIZATION ORIGINATING IT. POINTS OF VIEW OR OPINIONS STATED DO NOT NECESSARILY REPRESENT OFFICIAL NATIONAL INSTITUTE OF EDUCATION POSITION OR POLICY

Approved by:

Peter Calingaert
Adviser

W. A. Coulter, Jr.
Reader

David B. Benson
Reader

WILLIAM FRED BEYER, III. A Graphics System for Pole-Zero Map Analysis. (Under the direction of DR. PETER CALINGAERT.

ABSTRACT

An interactive, graphical display system has been developed for pole-zero map analysis. The facilities allow the user to specify a control system and an input function in the form of a pole-zero map and then examine the resulting transient response. The system is designed for use as an educational tool in teaching introductory courses in automatic control systems.

The hardware used is an IBM 2250 Model 1 Display Unit attached to an IBM System/360 Model 40 computer.

ACKNOWLEDGEMENTS

The author wishes to thank Dr. Peter Calingaert for his guidance in the preparation of this thesis and Dr. Norman Coulter for his review of the subject matter related to control theory.

Special thanks are also in order to Mr. James Batter for his assistance in the internal design of the system, Mr. James Whitlock for his sharp debugging eye, and Dr. Paul Oliver for his generous gift of sub-routines used in this system.

The IBM 2250 Display Unit was provided by the International Business Machines Corporation.

TABLE OF CONTENTS

	Page
CHAPTER 1 - INTRODUCTION	1
1.1 General	1
1.2 Objectives	3
1.3 Applications	5
CHAPTER 2 - BASIC CONCEPTS	7
2.1 Scope of the Analysis	7
2.2 Mathematical Models	9
2.3 Laplace Transform to Solve Linear Differential Equations	10
2.4 Transfer Function	14
2.5 Pole-Zero Map	15
2.6 The Algorithm	18
CHAPTER 3 - THE GRAPHICS DEVICE - IBM 2250	22
3.1 General Characteristics	22
3.2 Environment	25
3.3 Programming	26
3.4 Programming Support	27
CHAPTER 4 - SYSTEM USE	28
4.0 General	28
4.1 The Initial State-General Instructions Display	29
4.2 Setting Grid Parameters	34
4.3 The Pole-Zero Map Display	37
4.4 Displaying a Pole or Zero	39
4.5 Constant Multiplier	42
4.6 System Transfer Function vs. Input Response	43
4.7 Transient Response	45
4.8 Special Facilities	49
4.9 System Termination	51
4.10 Programmed Function Keys	51
4.11 Examples	56

Page

CHAPTER 5 - SOFTWARE DESCRIPTION 67

5.1 System Components 67
5.2 Important Data Structures 72
5.3 Core Requirements 73
5.4 Implementation 75
5.5 Hindsight 76

CHAPTER 6 - CONCLUSIONS 78

6.1 Evaluation of the System 78
6.2 Criticisms 83
6.3 Extensions 84

APPENDICES

A Laplace Transforms 85
B Sample Exercises 87
C Programmed Function Key Descriptions 90
D Turn-on and Turn-off Procedures for 2250 92
E UNC Hardware Configuration 93
F Computer Program Listings (separate binding) 94

Bibliography 95

LIST OF ILLUSTRATIONS

2-1	Electrical Circuit Diagram	8
2-2	Block Model with Transfer Function	14
2-3	An Example of a Pole-Zero Map	16
3-1	Photograph of IBM 2250 Model 1	23
3-2	IBM 2250 Hardware Environment	25
4-1	General Flowchart of System Use	30
4-2	General Instructions Display	32
4-3	Grid Parameter Display	35
4-4	Pole-Zero Map Display	38
4-5	Transient Response Display	46
4-6	Programmed Function Keyboard Plastic Overlay	52
4-7-A	Grid Parameter Display (example 1)	57
4-7-B	Pole-Zero Map with Inputs (example 2)	57
4-7-C	Transient Response to 4-8-B	58
4-7-D	Transient Response to 4-8-B with Component Curves	58
4-8-A	System Pole-Zero Map (example 3)	63
4-8-B	System Pole-Zero Map with Inputs	63
4-8-C	Transient Response to 4-8-B	64
4-8-D	Pole-Zero Map with a Multiple Pole	64
4-8-E	Transient Response of 4-8-D	65
4-8-F	Pole-Zero Map with Sinusoidal Input	65
4-8-G	Transient Response to 4-8-F	66

		Page
5-1	System Components	68
5-2	System Overlay Structure	74
6-1	Example of Unstable Response	80

I hear and I forget

I see and I remember

I do and I understand

-a Chinese proverb

Chapter 1

INTRODUCTION

1.1 General

PZMAP (Pole-Zero Map Analysis Program) is an IBM 2250 interactive graphics system for pole-zero map analysis. It is designed to serve as an educational tool for introductory courses in automatic control systems.

The basic facilities allow the user at the remote console to do the following:

1) Specify a linear control system by a pole-zero representation.
2) Specify an input function to this system by a pole-zero representation.
3) Investigate the properties of the transient (dynamic) response for this system over a given interval of time.

A large set of control systems may be examined including, with certain limitations, those with multiple poles and zeros. Capabilities also allow for the dissection of a transient response curve into its component curves. All related programming was done in IBM System/360 Assembler Language on the IBM System/360 Model 40 at the University of North Carolina Computation

Center.

An interactive graphics system implies that the user interacts with a computer via a graphical display device, in this case the IBM 2250 Model 1 (see Chapter 3 for description). The interaction is initiated by the user with various attention mechanisms such as a light pen, pushbutton keys and a typewriter-like keyboard. The functional operation of PZMAP is normally accomplished from one of two displays. One is the display of a pole-zero map grid and the other is of a transient response graph.

The user specifies a given control system by entering the coordinate positions of the associated poles and zeros from the alphanumeric keyboard. These are displayed on the pole-zero map grid by pushing a button. In a similar manner the user specifies an input function to the system. When the system specification is complete, the user initiates response computation by pushing another button and the transient response is then automatically plotted on the transient response display. Various edit and control facilities are also available. A detailed description of how to use the system is presented in Chapter 4.

1.2 Objectives

The general objectives are:

1) Maximize the set of acceptable pole-zero configurations.
2) Minimize the response time for computing transient responses, with an upper bound of ten seconds for extreme cases (in the IBM System/360 Model 40 environment).
3) Compromise the system design to provide a tool suitable for use in both a classroom environment and a laboratory environment.

The maximization of acceptable pole-zero configurations is intimately related to the minimization of the response times. More complicated systems require more computing to obtain the transient response and thus lengthen the response time. In an educational environment, fast response time is extremely important to maintain student interest. To satisfy this problem, an offset technique was utilized (described in Chapter 2, Section 6) which effectively sacrifices a degree of precision in the calculated response of complicated systems for a degree of speed in the response time.

The system design philosophy is influenced throughout by the criterion of serving both the classroom and the laboratory environments. These two

distinct educational situations are characterized as follows:

1) In the classroom situation a lecturer uses the graphics console as an integral part of the lecture to a seated audience.
2) In the laboratory situation an individual student, or a pair of students, work unsupervised while seated directly in front of the graphics console.

In many instances a requirement of one environment is in direct conflict with a requirement of the other environment. Consider, for example, the design of the display formats. In the classroom an important prerequisite is that every student may easily see the pole-zero maps and the transient response graphs from his classroom seat. This suggests making these displays as large as physically possible (the display area is 12" by 12"), thereby subordinating, or sacrificing, other desirable information on the display.

The laboratory environment poses no severe visibility problems because the user is seated in front of the console. What is required in this situation are displays which always provide sufficient operating instructions and hints for easy use. This is to eliminate frequent annoying references by the user to the accompanying user manual. This is important because most student users will be only

slightly familiar with the procedures of using the system. Experience has shown that the nuisance of referring to the user manual diminishes the enthusiasm of the student and the effectiveness of the system as a learning tool.[1]

This is one example of the divergence in the desirable characteristics of a system designed for both classroom and laboratory use. The displays for the classroom require large maps and graphs, whereas the laboratory requires abundant user instructions. Problems requiring compromise decisions also arise with other design considerations. How should a user enter data (keyboard vs. light pen)? How should initialization and operational procedures be designed (speed vs. ease of use)? What type of user information should be included (detailed vs. general)? The compromises made will be presented in the description of the system operation (Chapter 4).

1.3 Applications

The principal goal is to provide a system to be used as a teaching aid--not as a specific tool for solving problems. In particular, the system is

---

[1] A subjective appraisal on my part of an existing system [12] for teaching numerical analysis.

designed to illustrate the fundamentals of complex plane (s-plane) analysis. Such a tool is useful for presenting many basic concepts of control theory and application. Some of these, which will be illustrated later, are:

1) Transfer function
2) Pole-zero map
3) Transient response
4) Steady-state response
5) Series compensation
6) Stability
7) Damping

In the following chapters the system use and relevant background material are presented. Chapter 2 provides a review of the mathematical principles of elementary control theory and a formulation of the algorithm used in computing the transient response from the pole-zero configuration. Chapter 3 describes the IBM 2250 Model 1 graphics device. The system and its use are described in Chapter 4 with illustrations of the displays and examples. Chapter 5 discusses the computer program. In the concluding Chapter 6, an evaluation is made of the system in its current state and recommendations are presented for improvements. The appendices include all related program listings and additional program documentation.

Chapter 2

BASIC CONCEPTS

The purpose of this chapter is to present a very general review of, or introduction to, the mathematical principles of control systems analysis that were the basis for this graphics system. In the process, the algorithm for computing the dynamic response from the pole-zero map is developed.

2.1 Scope of the Analysis

The analysis of a control system involves the determination and evaluation of system behavior to an input signal. Various techniques and approaches have been developed. This discussion is limited to analysis by the Laplace transform with emphasis upon determining two factors:

1) the output response of a system as a function of time;
2) the relationship between system parameters and the output response.

Consider a simple electrical circuit (the system) consisting of an inductor L and a resistor R (the system parameters) as shown in Figure 2-1. The voltage, e(t), is the input signal and the current

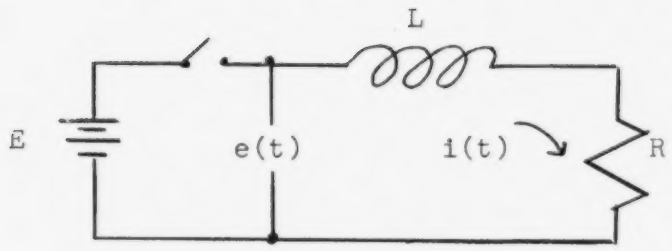


Figure 2-1

through the circuit, i(t), is the output response (both arbitrary). The system behavior as a function of time is described by i(t) for an input e(t). As will be seen, if e(t) is a step function of magnitude E starting at t = 0, the current through the circuit increases as a blunted exponential which approaches E/R with time constant L/R. Therefore, an increase in the resistance R or a decrease in the inductance L sharpens the rise of the exponential, whereas the opposite changes in R and L will slow the exponential rise.

The example of Figure 2-1 is used throughout this chapter to present the analysis principles. This is done purely as a convenience and is not to imply an applicability restricted to electrical systems. Suitable applications include any system which may be described by a mathematical model. This could be, for example, a physical system from the engineering

sciences, a physiological system from the biological sciences, or an abstract system.

2.2 Mathematical Models

The objective is to make a quantitative analysis of a given system. This analysis requires a mathematical model which imitates the behavior of the system as closely as possible. The mathematical model is constructed from a physical approximation to the system which makes necessary simplifying assumptions. Figure 2-1 is an approximate physical description of the actual circuit because certain assumptions are made concerning the system parameters. In this particular system, for example, it is assumed that the resistance is lumped in the resistor R. In the real circuit, however, there is resistance in the connecting wires, the inductor, and the power source, as well as the resistor. From this approximate system the mathematical model is derived.

Normally a mathematical model is constructed to determine the dynamic properties of the system. Therefore, the form is a differential equation or a system of differential equations. The differential equation (mathematical model), derived from basic electricity properties, describing the system of Figure 2-1 is

$$e(t) = Ri(t) + L\frac{di}{dt} \quad . \quad \text{(Equation 2-1)}$$

The problem now becomes how to solve such an equation, or to determine the function i(t) satisfying the equation. All differential equations in this thesis will be assumed linear with constant coefficients.

2.3 Laplace Transform to Solve Linear Differential Equations

The Laplace transform approach to solving linear equations is an operational method which yields a solution algebraically. It is commonly used with control systems because of the analysis techniques which are a natural result of the method. Also, for the complicated engineering systems encountered, this method is usually easier than the classical analytical methods for solving differential equations.

The basic operation transforms a function of time t into a function of a complex variable s. The formal definition [4] is:

> Let f(t) be a function of a real variable t defined for t > 0. Then the Laplace transform of f(t), denoted $\mathcal{L}\{f(t)\}$, is

$$\mathcal{L}\{f(t)\} = F(s) = \int_{\epsilon}^{T} \lim_{T\to\infty} f(t)e^{-st}dt$$

$$\epsilon\to 0^+$$

$$= \int_{0^+}^{\infty} (f(t)e^{-st}dt$$

where $0 < \epsilon < T$, s a complex variable defined by $s = \sigma + j\omega$, σ, ω real variables and $j^2 = -1$. See Appendix A for a table of common transforms.

The general method of solution involves transforming each term of the differential equation to the complex s domain, rearranging the resulting equation so as to isolate the dependent variable, and then taking the inverse transform to obtain a direct solution. The inverse operation is defined formally in Appendix A though the operation is normally performed by utilizing available tables of transforms.

Consider Equation 2-1. The Laplace transform of each term, with the aid of the tables in Appendix A, yields the equation

$$E(s) = RI(s) + LsI(s) - i(0^+). \qquad \text{(Equation 2-2)}$$

By isolating I(s) and assuming initial conditions are zero, the result is

$$I(s) = E(s)\frac{(1/L)}{s + (R/L)} \quad . \qquad \text{(Equation 2-3)}$$

The inverse Laplace transform of this equation will produce the current description, i(t).

The determination of the inverse transform is generally not easy if the form is not in the available tables. Consider the general case where the output response x(t) of a system is desired. The transformation to the s-plane and the isolation of X(s) will

always produce a ratio of two polynomials in s of the form

$$X(s) = \frac{K(s+Z_1)(s+Z_2)\cdots(s+Z_m)}{(s+P_1)(s+P_2)\cdots(s+P_n)} \qquad \text{(Equation 2-4)}$$

where K is a constant multiplier and P_i and Z_i are roots (real or complex) of the polynomials (note that Equation 2-3 is of this form). If X(s) represents a system with n,m of high order, the form will not be listed in common tables of Laplace transforms and computation of the inverse is nontrivial.

In order to simplify this problem, a normal method of solution is to expand the function by partial fractions. This results in a sum of simpler terms multiplied by constants. These individual terms are easily transformed into the main domain and the total response is the sum of the individual responses. This summation is possible because of the linearity assumption which implies the law of superposition. The general expansion of Equation 2-4 <u>when all of the P's are distinct and m < n</u> is

$$X(s) = \frac{K_1}{s+P_1} + \frac{K_2}{s+P_2} + \cdots + \frac{K_n}{s+P_n} \qquad \text{(Equation 2-5)}$$

where K_i is the <u>residue</u> of X(s) at $s = -P_i$. The case when $n \leq m$ will not be considered. The problems resulting when the P's are not distinct (multiple poles) will be discussed in Section 2.6.

The partial fraction expansion of Equation 2-3 with $E(s) = \frac{E}{s}$ (the Laplace transform of a step function of magnitude E) is

$$I(s) = \frac{E\ (1/L)}{s(s+(R/L))} = \frac{E}{R}s - \frac{E}{R(s+(R/L))}$$

$$= \frac{E}{R}\left(\frac{1}{s} - \frac{1}{s+(R/L)}\right) \quad . \qquad \text{(Equation 2-6)}$$

There are two residues (both real) for I(s): E/R at s = 0 and E/R at s = -R/L. By taking the inverse Laplace transform of each individual term, the description of i(t) is obtained as follows:

$$\mathcal{L}^{-1}\{I(s)\} = \frac{E}{R}\left(\mathcal{L}^{-1}\left\{\frac{1}{s}\right\} - \mathcal{L}^{-1}\left\{\frac{1}{s+(R/L)}\right\}\right) \qquad \text{(Equation 2-7)}$$

$$i(t) = \frac{E}{R}(1 - e^{-(R/L)t}) \qquad \text{(Equation 2-8)}$$

Therefore, the system behavior for a step function of magnitude E starting at t = 0 is composed of two separate subfunctions: a constant function and a decaying exponential of time constant L/R, i.e.,

$$i(t) = i_1(t) + i_2(t) \qquad \text{(Equation 2-9)}$$

where $i_1(t) = E/R$ and $i_2(t) = -(E/R)e^{-(R/L)t}$. The output function, i(t), is the dynamic response of the system.

The Laplace transform technique is a straightforward approach which lends itself to engineering applications, as opposed to the classical methods which are based on guess-work. More importantly, this

approach leads to useful tools for analysis. These tools, to be described in the next sections, are "by-products" of the method.

2.4 Transfer Function

The transfer function of a system, denoted G(s), is defined as the ratio of the Laplace transform of the output function to the Laplace transform of the input function with initial conditions assumed to be zero. This is an important relationship because it relates the output response to the input response in a manner which is independent of the internal system structure. Once again consider Equation 2-2 with zero initial conditions. Rearrange the equation as follows:

$$G(s) = \frac{I(s)}{E(s)} = \frac{1}{R+Ls} = \frac{1/L}{s+(R/L)} \qquad \text{(Equation 2-10)}$$

This is the transfer function of the electrical circuit discussed earlier. An appropriate block model for this system would be that of Figure 2-2. Notice that the equation for I(s) is obtained by multiplying

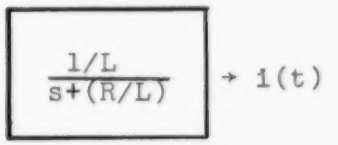


Figure 2-2

Block Model Using Transform Function

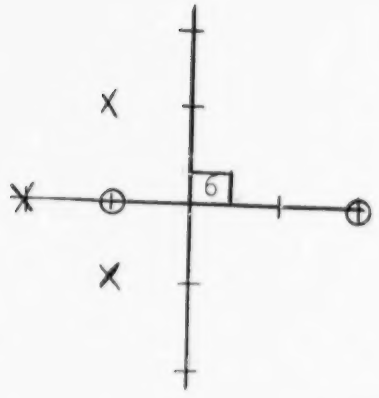
the Laplace transform of the input by the transfer function, i.e., $I(s) = E(s)G(s)$.

The transfer function is a useful tool in the analysis and design of control systems. It is a direct result of the Laplace transform method and it provides a mathematical model of the dynamic properties of the system. The utility of the transfer function will be illustrated in the next section.

2.5 Pole-Zero Map

A zero of the transfer function is a value of s for which $G(s) = 0$ and a pole is a value of s for which $G(s) = \infty$. With G(s) in the form of Equation 2-4, the zeros are the roots of the numerator polynomial and the poles are the roots of the denominator polynomial.

A graphical s-plane representation of the poles and zeros of the system transfer function is called a pole-zero map. An example is shown in Figure 2-3. The poles are represented by X's and the zeros by O's. The box in the right half-plane contains the constant multiplier which is a pure magnitude factor. Note that the pole-zero map and the equation of G(s) represent the same amount of information. Either is a complete representation of



$$G(s) = \frac{6(s+1)(s-2)}{(s+2)(s+1+j)(s+1-j)}$$

Figure 2-3

An Example of a Pole-Zero Map

the system transfer function.

Multiple poles and zeros (more than one pole or zero at the same point in the s-plane) are normally represented by slightly offsetting the symbols. In this graphics system they are designated by an exponent appended to the symbol.

Examination of the pole-zero map can yield information about the system behavior. From the partial fraction expansion in Equation 2-5, it is seen that there exists a direct relationship between the pole locations and the form of the system response.

Furthermore, the relative influence of a pole upon the total response is dependent upon the mangitude of its associated residues. The zeros indirectly affect the response by influencing these residues.

From the above observations and the table of Laplace transforms (Appendix A), it may be seen that the time response of a system is composed of basically simple functions. These are dependent upon the poles and occur as follows:

1) a pole at the origin, $s = 0$, contributes a step function of some magnitude K;
2) a pole on the real axis, $s = b$, contributes an exponential response of the form Ke^{bt};
3) a pair of complex conjugate poles on the jw-axis, $s = \pm jw$, contribute a sinusoid of the form $K \sin(wt)$;
4) a pair of complex conjugate poles, $s = b \pm jw$, where $b, w \neq 0$, contribute a damped sinusoid of the form $Ke^{bt}\sin(wt + \emptyset)$ where $\emptyset$ is a phrase shift (this decays or grows within an exponential envelope).

It should be recognized here than an input function has poles and zeros associated with it in the Laplace domain just as the system transfer function

does. For example, a step function in the s-plane is a pole at the origin and sinusoid is a pair of complex poles on the jw-axis. Therefore, the complete pole-zero description of a system must include the poles and zeros of both the transfer function and the input function. This is verified from the construction of $G(s) = X_0(s)/X_1(s)$ which yields the output function $X_0(s) = G(s)X_1(s)$.

2.6 The Algorithm

The graphic capabilities of this system display the dynamic response from a pole-zero description of a specified system. This description includes the system transfer function, the Laplace transform of the input function and the constant multiplier, all specified by the user at the graphics console. Therefore, the major computational problem is to obtain the inverse Laplace transform from a vector of poles and zeros, and adjust the magnitude of the response according to the constant multiplier.

In Section 2.3 it was shown that the response could be obtained from a partial fraction approach. A formal statement of the algorithm used is:

Given: a vector of poles $P = (P_1, P_2, \cdots, P_n)$
and a vector of zeros $Z = (Z_1, Z_2, \cdots, Z_m)$.

The equivalent function R in s is

$$R(s) = \frac{(s-Z_1)(s-Z_2)\cdots(s-Z_m)}{(s-P_1)(s-P_2)\cdots(s-P_n)} .$$

There exists a partial fraction expansion for R(s) if n > m. Furthermore, assume each element of P is distinct ($P_i \neq P_j$ when $i \neq j$). This partial fraction expansion is

$$R(s) = \sum_{i=1}^{n} \frac{K_i}{(s-P_i)}$$

where the residue K_i of R(s) at $s = P_i$ is given by

$$K_i = \lim_{s \to P_i} (s-P_i)R(s) .$$

Note that R(s) is more than the system transfer function because the poles and/or zeros of the input function are included.

The inverse Laplace transform, r(t), or R(s) is

$$r(t) = \mathcal{L}^{-1}\left\{\sum_{i=1}^{n} \frac{K_i}{s-P_i}\right\}$$

$$= \sum_{i=1}^{n} K_i \mathcal{L}^{-1}\left\{\frac{1}{s-P_i}\right\} .$$

The fact that the poles are the roots of a polynomial with real coefficients implies that any complex poles will occur as conjugate pairs. Therefore, the system response may be obtained from transforms 4 and 8 of Appendix A depending upon whether each pole is real

or complex:

If $P_i = a$ is real, then

$$K_i \mathcal{L}^{-1}\left\{\frac{1}{s-a}\right\} \Longrightarrow K_i e^{at} \qquad \text{(Transform 4)}$$

If $P_i = a + \omega j$ is complex,

let $P_{i+1} = \overline{P}_i = a - \omega j$ (complex conjugate)

and let $K_i = c + dj$ which implies

$K_{i+1} = \overline{K}_i = c - dj$

then

$$K_i \mathcal{L}^{-1}\left\{\frac{1}{s-P_i}\right\} + \overline{K}_i \mathcal{L}^{-1}\left\{\frac{1}{s-\overline{P}_i}\right\} \Longrightarrow 2|K|e^{at}\sin(\omega t+\emptyset)$$

where $K = \sqrt{c^2 + d^2}$ and $\emptyset = \tan^{-1}(c/d)$.

(Transform 8)

The restriction that every pole by distinct is severe and must be considered. The effect of a pole P_i of multiplicity $q > 1$ is to introduce a term of the form $(s-P_i)^q$ in the denominator of R(s). This significantly increases the complexity of the partial fraction expansion by necessitating the computation of derivatives and additional residues.

To allow the use of multiple poles, and yet circumvent the inherent computational complexities, an offset technique is used. The basic premise is that, for example, $(s-P_i)^2 \simeq (s-P_i)(s-P_i + \epsilon)$ where ϵ is small and, therefore, any resulting difference in behavior will be slight. A pole P_i of multiplicity

$q > 1$ is considered to be a sequence of q poles at $P_i + k\epsilon$ where $k = 0,1,-1,2,-2,\cdots$. The effect is to compute q distinct residues for q distinct poles. The response associated with the multiple pole is the sum of these individual responses. The constant value of ϵ is .011.

The value is large enough to avoid severe round-off error yet small enough not to change appreciably the response of the system. See Section 6.1 for a discussion of the effect on accuracy. The exact offset procedure is:

> Given P_i of multiplicity $q > 1$, let $(P_i)_k$ be the k^{th} offset pole where $(P_i)_0 = P_i$. Then $(P_i)_k = P_i + (-1)^{k+1}\lceil k/2 \rceil (.011)$.

In the case where P_i is complex, only the real part is offset.

Chapter 3

THE GRAPHICS DEVICE - IBM 2250

3.1 General Characteristics

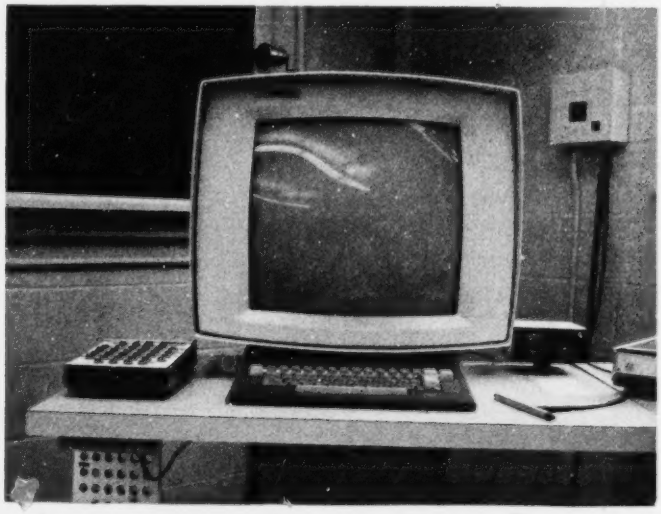
The IBM 2250 Model 1 Display Unit (Figure 3-1) is a remote graphics terminal device designed to expand the scope of the IBM System/360. The unit is programmable and may run concurrently with the System/360 by means of an 8K (8 x 1024 bytes) buffer. The principal component is a cathode ray tube (CRT) with capabilities for the high speed display of points, vectors and alphanumeric characters. Communication between the user and the system is accomplished by manual input attention mechanisms which initiate system interrupts. Available attention mechanisms are the alphanumeric keyboard, light pen, and thirty-two pushbutton keys on the programmed function keyboard. An audible alarm is provided to signal the user.

Display Area

The display area is a 12" by 12" CRT. This area is divided into 4096 by 4096 raster units with 1024 by 1024 addressable points identified by x- and y-coordinates. This provides approximately one million addressable intersection points and 3,848 addressable

character positions (74 characters per line on 52 lines).

An image is created by bombarding the CRT with a controlled electron beam from a cathode, in either a character or vector mode. The beam is controlled by the sequential execution of graphics orders in the buffer. The image from the beam upon the CRT fades within a fraction of a second. To produce a continuous image, the graphic orders are executed (the image is regenerated) approximately 40 times per second.



Photograph of IBM 2250

Figure 3-1

Buffer

The 8K buffer storage is local to the display unit and is used to store graphic orders for image regeneration. The images are stored in the form of a sequence of graphic orders which control the position of the electron beam. This allows concurrency of operations between the display unit and the computer.

Alphanumeric Keyboard

This is a typewriter-like keyboard which allows the user to enter characters on the CRT. There are 44 keys and a space bar which represent 63 standard characters. The position at which a character will be displayed is designated by a cursor which is an underbar symbol directly beneath the character position. The cursor is positioned by programming and/or keyboard controls.

Programmed Function Keyboard

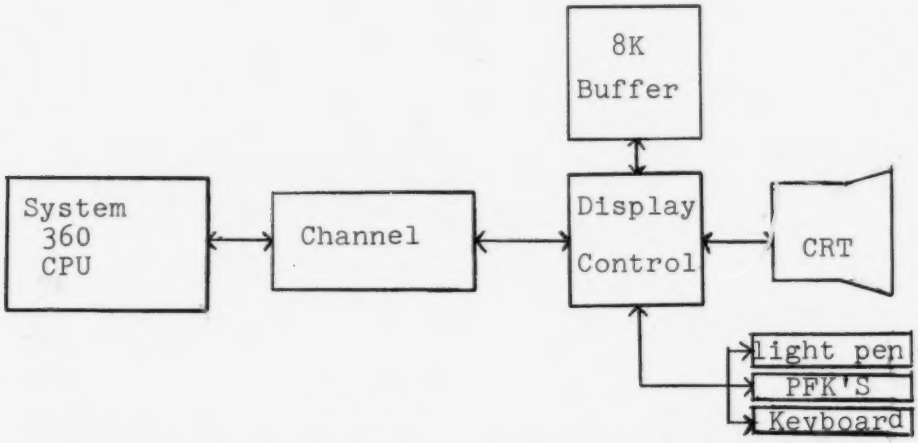
The programmed function keyboard is a panel consisting of 32 programmed function keys (PFK's) and 32 indicator lights. Plastic overlay covers (see Figure 4-6) which set 8 sensing switches on the panel, allow for a possible 256 distinct function key configurations. The function of each PFK is defined within the executing program.

Light Pen

The light pen is a pen-like device containing a photocell element which enables the display unit user to designate a position upon the CRT. It is used by pointing the pen at the CRT image to be recognized. Light emitted from the CRT is recognized by the pen, causing a system interrupt which transmits position information to the executing program.

3.2 Environment

A general schematic of an IBM System/360 environment with an IBM 2250 Model 1 is shown in Figure 3-2 (see Appendix E for detailed schematic of the UNC Computation Center configuration). The 2250 is connected to the selector channel by a standard I/O interface.



A Schematic Diagram of an IBM System/360 Environment with an IBM 2250 Display Unit

Figure 3-2

3.3 Programming

Graphic programming services are available to handle the following functions:

1) graphic data management;

2) input/output control;

3) attention handling.

The 2250 is programmed by using these features in conjunction with a host programming language (IBM System/360 Assembler Language or a higher-level language). A 2250 graphics program is executed in main memory and controls the communication with the 2250.

Consider the programming to create an image upon the CRT. The image is produced by the continuous sequential execution of a buffer program which controls the electron beam. This buffer program is built in the graphic data output area (GDOA) in the 2250 graphic program and then passed by means of an output instruction to the buffer via the channel. When the buffer (beam control) program is in the buffer, a command from the graphic program will start execution of the buffer program to create an image upon the CRT.

The executions of the graphics program in main memory and a buffer program in the buffer are independent. An attention inquiry facility allows the graphic program to enter a wait state until an attention (light pen, PFK, etc.) is recognized. This frees the central

processing unit (CPU) for other uses while the buffer program is independently regenerating an image upon the CRT. When an attention is recognized, the graphic program seeks control of the CPU and starts execution from the point of the wait state. By analyzing the attention code and branching accordingly, different actions may be taken for different attentions.

3.4 Programming Support

Currently IBM has available several levels of 2250 programming services for use with OS/360. At the assembler language level, there are a set of macro instructions complemented by a set of problem oriented routines (POR's) which produce graphic orders for common image types. For 2250 use with higher level languages, graphic subroutine packages are available for FORTRAN and PL/1 programs.

Chapter 4

SYSTEM USE

This chapter discusses how to use the system. It is designed to serve both as a user manual and as part of the system documentation. Only the general aspects of the system are presented. Any items particular to the UNC implementation are described in Appendix D. This includes turning the device on and off and initiating system execution. This chapter assumes the user is familiar with terms discussed in Chapter 2 and the device discussed in Chapter 3.

4.0 General

The basic facilities allow the user to examine the transient response (and its components) of a given system specified in pole-zero form for a given input specified in pole-zero form. This is accomplished by the manipulation of four distinct displays:

1) general instruction display;
2) grid parameter display;
3) pole-zero map display;
4) transient response display.

The manipulation is done by use of the programmed function keys (PFK's), the alphanumeric keyboard and the light pen.

The user is either sitting directly in front of the 2250 if in a laboratory situation, or standing beside the 2250 if in a classroom lecture situation. System operation is the same in either case.

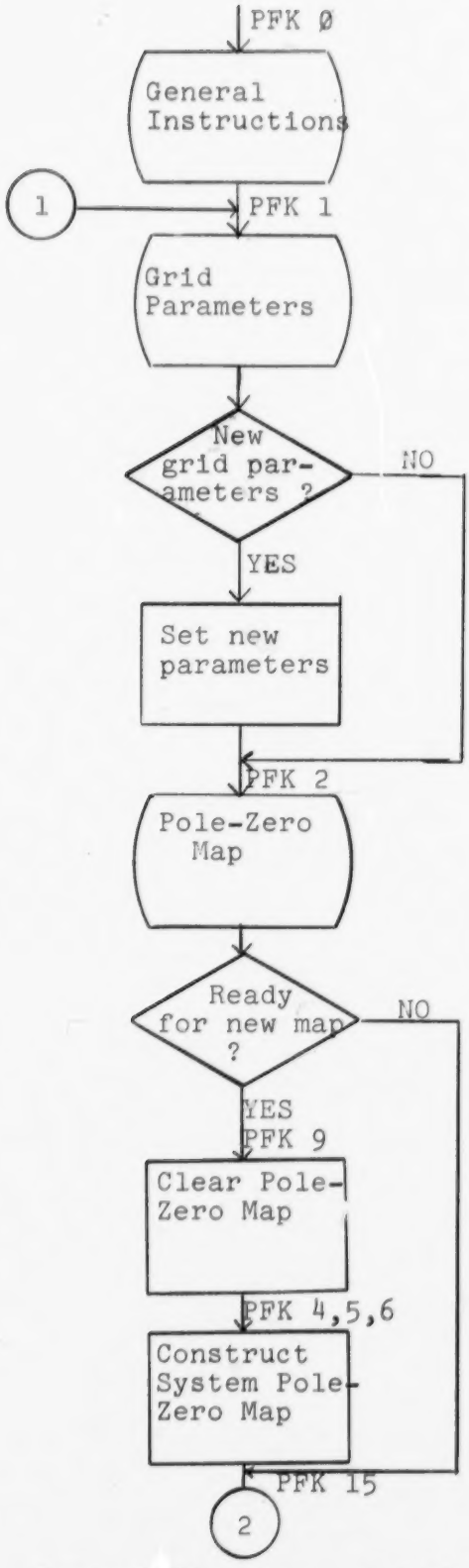
To operate the system, the user should follow a general sequence of steps:

1) set the grid parameters;
2) display the pole-zero map;
3) construct the pole-zero map of the system transfer function;
4) specify the input function for the system;
5) compute and display the transient response;
6) re-display the system transfer function and proceed from step 4).

Editing facilities are provided for the displays as are other features to facilitate easy use. Figure 4-1 presents a general flowchart for using the system.

4.1 The Initial State - General Instructions Display

The initial display (Figure 4-2) is a list of general instructions showing the active PFK's (0 - 16) and a brief description of the function of each. At this point the user should insure that the plastic overlay 0 is properly inserted over the programmed function keyboard. Otherwise, the system is completely inactive. To proceed from this point, depress PFK 1 for the grid



General Flowchart of System Operation

Figure 4-1

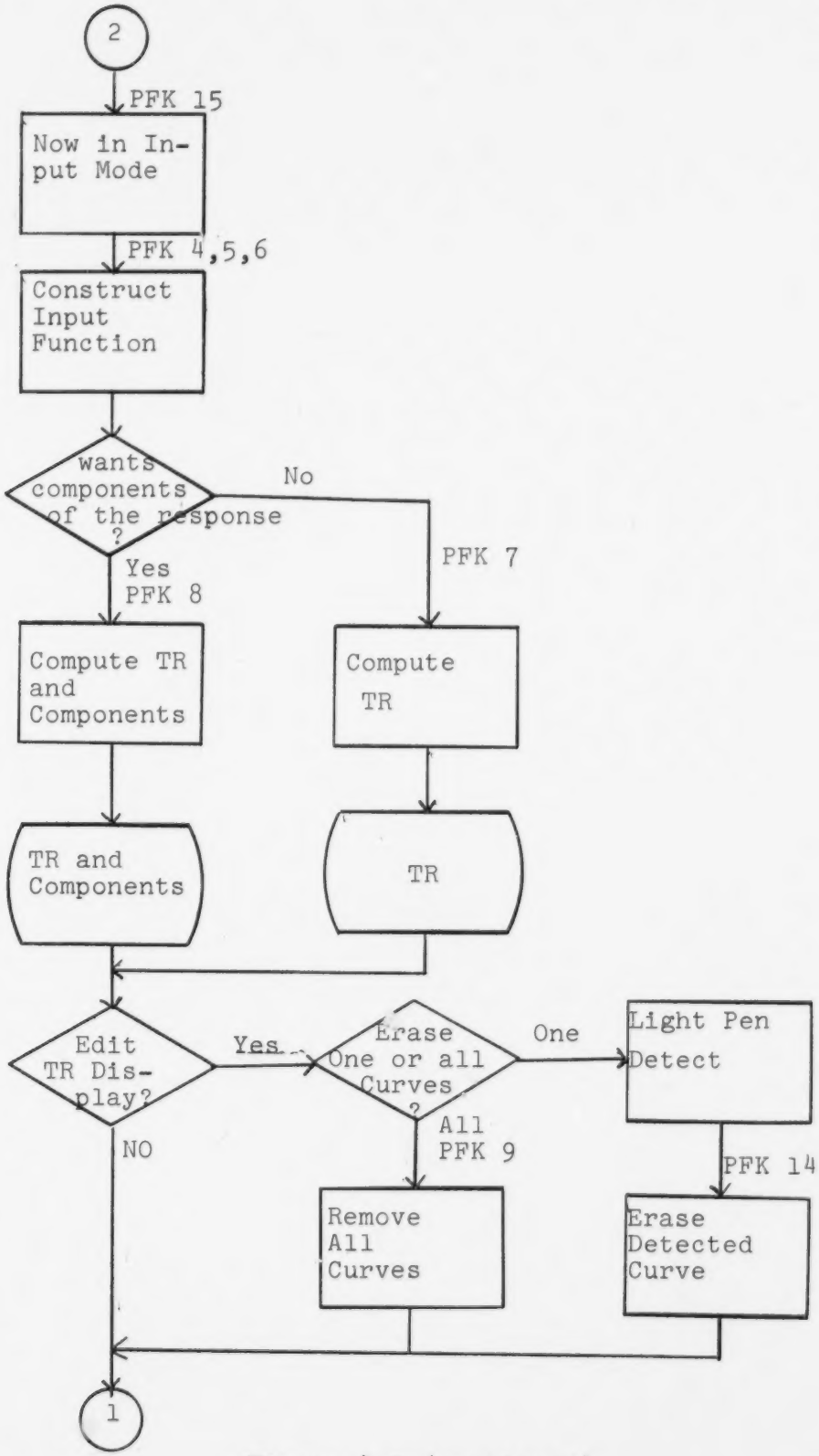


Figure 4-1 (continued)

POLE-ZERO MAP ANALYSIS

GENERAL INSTRUCTIONS

PFK 00-DISPLAY THESE INSTRUCTIONS
PFK 01-DISPLAY TO SET GRID PARAMETERS
PFK 02-DISPLAY POLE-ZERO MAP
PFK 03-DISPLAY TRANSIENT RESPONSE
PFK 04-PLOT ZEROS
PFK 05-PLOT POLES
PFK 06-ENTER CONSTANT MULTIPLIER
PFK 07-COMPUTE AND DISPLAY THE TRANSIENT RESPONSE
PFK 08-COMPUTE COMPONENTS OF TRANSIENT RESPONSE
PFK 09-CLEAR CURRENT GRID
PFK 10-REMOVE COMPONENT GRAPHS
PFK 11-STEP INPUT
PFK 12-RAMP INPUT
PFK 13-SINUSOIDAL INPUT
PFK 14-ERASE A DETECTED CURVE
PFK 15-DESIGNATE INPUT MODE
PFK 16-PLACE CURSOR
PFK 31-TERMINATE PROGRAM

TO CONTINUE - PRESS PFK 1

General Instructions Dislay (PFK 0)

Figure 4-2

parameter display. All other PFK's are inactive until this is done.

The general instructions may be generated at any time during system operation by depressing PFK 0. This re-initializes the system to the original state including clearing all displays. PFK 0 should be used as a recovery mechanism whenever the user is irretrievably lost. A complete list of PFK's with a description of their related functions is included in Section 4.10.

Certain errors result in messages printed out on the console of the IBM System/360. These messages are relayed to the user by the computer operator. These messages with appropriate user action are:

1) "2250 USER HAS WRONG OVERLAY"

The wrong plastic overlay is being used with the programmed function keyboard. Insert the correct overlay or insure that the overlay is properly inserted over the keyboard.

2) "2250 SPAR HAS FAILED"

The program attention routines were not properly linked to the system. Ask the computer operator to clear the partition and to release the program again.

3) "BUFFER ALLOCATION ERROR"

The buffer was not properly allocated. Ask the computer operator to clear the partition and to release the program again.

4) "POLE-ZERO 2250 PROGRAM GO"

Initialization procedures have been successful. Proceed.

5) "I/O ERROR ON 2250"

An I/O error occurred during communication between the 2250 and the computer. The general instructions are automatically displayed. Proceed from this point. If the error persists, notify the computer center personnel.

4.2 Setting Grid Parameters

Grid parameters must be established for the pole-zero map grid and the transient response graph grid. This is done by the user via the alphanumeric keyboard from the grid parameter display (Figure 4-3) generated by depressing PFK 1. The initial default conditions are those shown in Figure 4-3. If these are satisfactory proceed by depressing PFK 2.

The Grid Parameter Display

The grid parameter display consists of the grid parameter fields and a message area plus brief instructions. The grid parameter fields consist of twelve 6-character fields with three fields each for the σ and $j\omega$ axes of the pole-zero map and the x and y axes of the transient response graph. The leftmost fields on the display are the lower bounds of the respective axes, the TO fields are the upper bounds and the BY fields are the increments between coordinate lines perpendicular to the axis. For example, Figure 4-7-C shows the transient response grid with the x-axis grid parameters set as follows: " 0 TO 5 BY 1". Note the vertical lines at increments of one as specified in the BY field.

ENTER GRID PARAMETERS

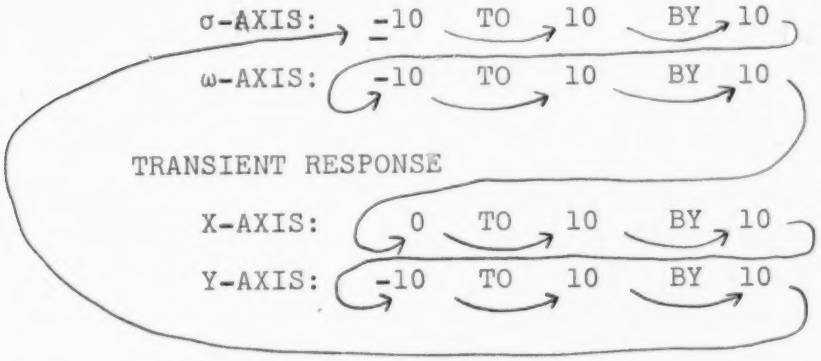
THE CURRENT PARAMETERS FOR THE
POLE ZERO MAP AND THE TRANSIENT
RESPONSE GRAPH ARE DISPLAYED BELOW

TO CHANGE THESE - USE THE KEYBOARD

WHEN FINISHED - PROCEED BY PRESSING PFK 2

MSG -

POLE-ZERO MAP



The Grid Parameter Display

Figure 4-3

Entering the Grid Parameters

The cursor is initially positioned under the first character of the leftmost field of the σ-axis fields. To enter data in this field, use the alpha-numeric keyboard. Any integer or fixed-decimal format is valid, but a floating point notation is not. The range of permissible grid values is limited only by the field size of six characters.

The cursor is advanced from one field to the next by depressing the JUMP key located at the upper left of the alphanumeric keyboard. The path of the cursor is illustrated by the arrows of Figure 4-3. The PLACE function key (PFK 16) will alternately set the cursor to the first field of the transient response fields and the pole-zero map fields.

The Message Area

The 36-character message area is for the display of all error messages related to setting the grid parameters. Any message is always accompanied by the sounding of the audible alarm. These error messages are not displayed until an attempt is made to display one of the grids with PFK 2 or 3. Consequently, the new grid parameters are not updated until one of the two grids has been successfully displayed. Possible error messages with the appropriate action are:

1) "ERROR IN PZ GRID PARAMETERS"

An error has been found in the pole-zero map grid parameters. Check the pole-zero grid fields and correct any errors - then try again. An example would be a lower grid bound greater than the upper bound.

2) "ERROR IN TR GRID PARAMETERS"

An error has been found in the transient response grid parameters. Similar to the previous message.

3) "BY PARAMETER IS TOO SMALL"

A parameter specified in a BY field is too small requiring too much buffer space to display the grid. Increase the size of this parameter and try again.

4) "ERROR IN ZEROS"

An error has occurred in plotting the current zeros with the new parameters. Either remove the zeros or increase the range of the parameters appropriately.

5) "ERROR IN POLES"

An error has occurred in plotting the current poles. Similar to the previous message.

6) "CONVERSION ERROR"

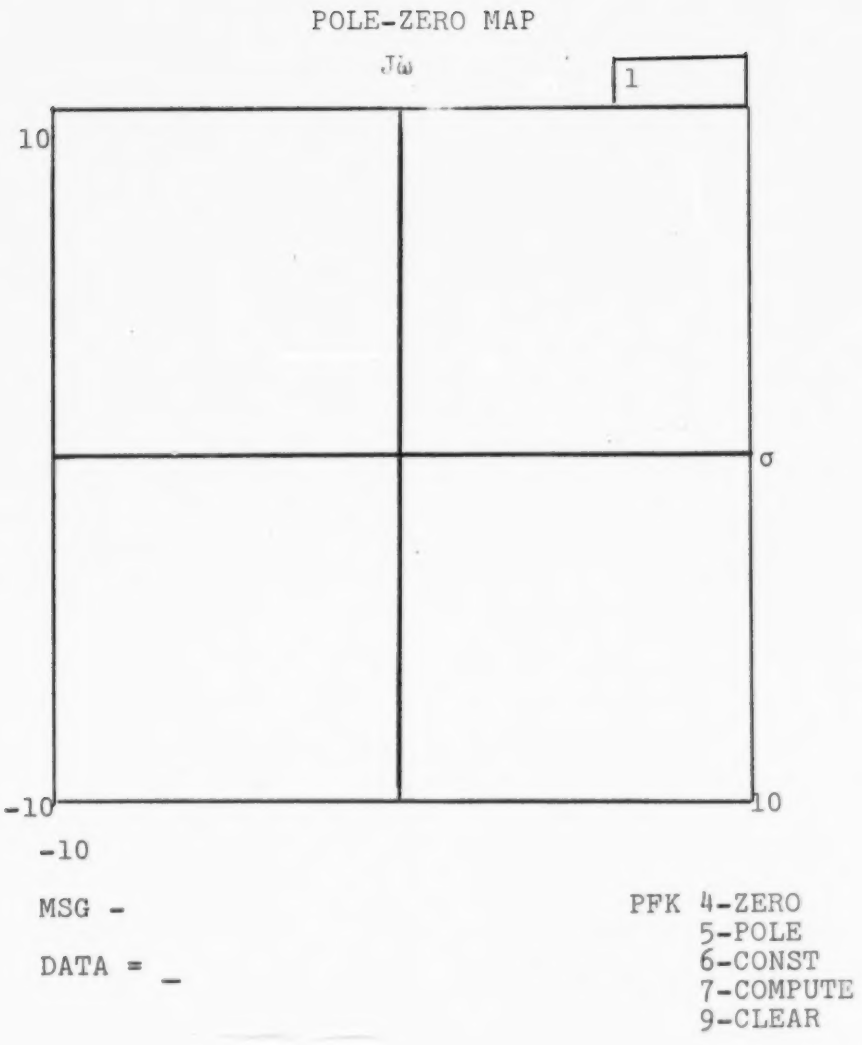
An error has occurred in converting the current grid parameters to floating point. Remove any nonnumeric data and try again.

7) "INVALID FUNCTION KEY"

The function key used has no meaning with the current display.

4.3 The Pole-Zero Map Display

The pole-zero map display (Figure 4-4) is generated from PFK 2 according to the current grid parameters established in the grid parameter display. The basic parts of the display are the grid with appropriate labels for the grid bounds, the 36-character



Pole-Zero Map Display

Figure 4-4

data pad (DATA = _ ...) for keyboard input, the 36-character message area (MSG - ...) for error and warning messages and the constant multiplier box located in the upper right corner of the grid (initially set

to one). A menu of some of the relevant PFK's is also provided. The cursor is positioned to the first character of the data pad.

4.4 Displaying a Pole or Zero

Poles or zeros (henceforth poles/zeros) are displayed upon the pole-zero display by specifying the coordinate positions in the data pad and then depressing PFK 5 or 4 respectively. Consequently poles and zeros may be displayed only in separate steps. Zeros appear as the symbol O and poles appear as the symbol X (see Figure 4-8-A). Multiple poles/zeros are designated by an appended exponent to the X/O showing the multiplicity (see Figure 4-8-D).

The input string in the data pad must provide the σ and $j\omega$ coordinates for each pole/zero to be displayed. These are specified according to simple rules of syntax. Each entry must be delimited by a comma and the string must terminate with a simicolon.

Real Pole/Zero Entry

For specification of a real pole/zero, only the real part is entered into the input string.

Complex Pole/Zero Entry

For specification of a complex pole/zero, the real and imaginary parts are separated by

a comma and enclosed in parentheses. Only one of the complex conjugate pair need be specified though both will automatically be displayed. If the conjugate is specified also, it is interpreted as a multiple pole/zero entry.

Examples

Input string	Specified pole/zero positions
1;	(1,0)
-2,-3;	(-2,0),(-3,0)
(1,2);	(1,±2j)
(0,4),(-1,1);	(0,±4j),(-1,±j)
-6,(4,2);	(-6,0),(4,±2j)
(-1.8,-8),0;	(-1.8,±8j),(0,0)

All entries must be in integer or fixed decimal format (no floating point notation accepted). Blanks are admitted but ignored. The range of permissible values is limited by the current grid parameter bounds. No more than ten distinct poles and zeros may be displayed on the map at any time and the multiplicity of any pole/zero may not exceed nine. The response to systems with pole multiplicities greater than three, however, results in computation interrupts and may not be displayed.

Error Messages

If an error is detected in the data pad input string after depressing PFK 4 or 5, an appropriate

6) "CONVERSION ERROR"

An error occurred in converting a coordinate specified in the data pad to floating point. Correct the error and continue. Probably an alphabetic character has been included.

7) "NOW IN INPUT MODE"

A warning message to indicate that the system is now in the input mode until the pole-zero map is displayed again from PFK 2.

8) "INVALID FUNCTION KEY"

The function key used has no meaning with the present display.

4.5 The Constant Multiplier

The current constant multiplier is located in the rectangular box in the upper right corner of the pole-zero map display. It is initially set to 1 (see Figure 4-4). To specify a new constant multiplier, enter the value in the data pad, terminate the string with a semicolon and depress PFK 6 (CONST). The new constant is copied into the box right justified. The constant must be in integer or fixed decimal format.

Examples

Input String	Constant Multiplier
1000;	1000
1000.;	1000
1000.5,	1000.5
.20;	.2

Error Messages

Error messages are displayed in a manner similar to the pole/zero entries. Note that for constant multipliers only one entry is valid in the data pad. Blanks are admitted but ignored.

In the input mode (to be described in Section 4.6) the new constant multiplier is the product of the current multiplier and the entry in the data pad.

4.6 System Transfer Function vs. Input Function

Some distinction must be made between entries for the system transfer function and entries for the input function. Therefore, poles, zeros and constant multipliers are entered from one of two modes: the system mode and the input mode. All entries from the system mode are considered as part of the system transfer function and all entries from the input mode are considered as part of the input function.

Whenever the pole-zero map display (PFK 2) is generated, the system mode is automatically in effect. If a pole-zero map has been created previously, only those entries associated with the system transfer function will be displayed. All new entries (poles, zeros and constant multipliers) are part of the system transfer function until PFK 15 (INPUT) is depressed. This initiates the input mode by sounding the audible alarm and displaying the message "NOT IN INPUT MODE" in the message area. The pole-zero map on display at this time is the specification of the system transfer function.

In the input mode all entries are considered as part of the input function specification. The poles and zeros are entered in the exact manner as in system mode and no physical distinction is made between the two types on the display. The constant multiplier is also entered in the same manner; however, the effect is slightly different because the response function is the product of the system transfer function and the input function. The new constant multiplier will be the product of the current multiplier and the entry in the data pad. From the system mode, the data pad entry is directly copied into the constant multiplier box.

The input mode is terminated automatically whenever the pole-zero map display is regenerated with

PFK 2. At this time the system transfer function is displayed and input elements are eliminated. To examine the response of this system to another input, depress PFK 15 and then specify the new input function.

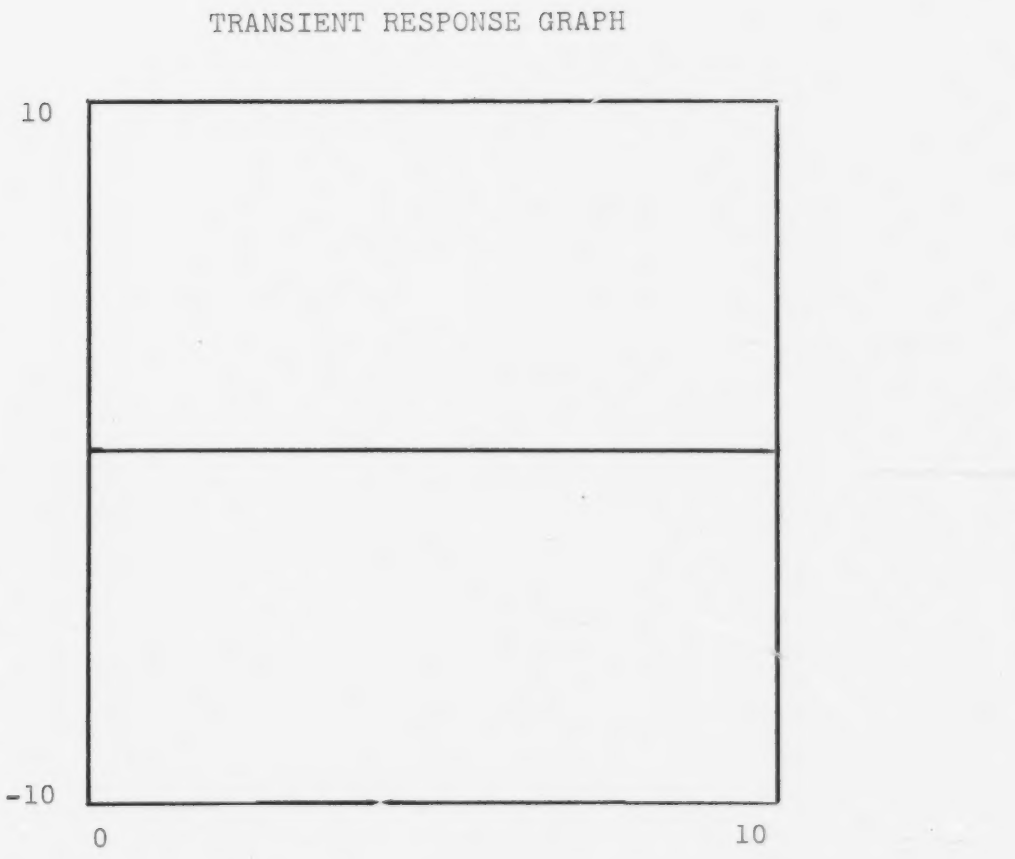
4.7 The Transient Response

The Display

PFK 3(TRESP) generates the transient response display (Figure 4-5) and any curves present from previous computations. The display is similar to the pole-zero map display. There is a grid with axes labeled according to the current grid parameters, a 36-character message area and a menu of some of the PFK's relevant to this display. There are no manual input facilities. The use of PFK 3 only generates the display and does not evoke any response computations.

Computing the Response

The transient response of the current pole-zero configuration may be computed only from the pole-zero map display. This is done by using either PFK 7 (TR) or PFK 8 (TRCMP). PFK 7 displays the total system response. PFK 8 displays the total system response and the component curves which make up this response. A discussion of this is provided in Section 2.6. This system cannot compute the response of a pole-zero configuration where the number of zeros is equal to or greater than



Transient Response Display

Figure 4-5

the number of poles.

When PFK 7 or 8 is depressed, the transient response display is immediately generated along with any previous response curves (same effect as PFK 3). The display of the new response curve(s) associated with

the current pole-zero configuration is generated from one to ten seconds later depending upon the complexity. There is a momentary fade of the display when this occurs. If no curve is displayed after this momentary fade, adjust the grid parameters and try again.

Erasing Component Curves

The component curves resulting on the transient response display when PFK 8 is used may be removed by depressing PFK 10 (CMPERSE). The total transient response curve remains.

Error Messages

If an error occurs during the computation of the response, an appropriate error message is displayed in the message area of the transient response display and the audible alarm sounds. When this happens, return to the pole-zero map or grid parameter display and correct the error source. Possible error messages and appropriate action are:

1) "INVALID TRANSFER FUNCTION"

The total number of zeros in the pole-zero map is equal to or greater than the number of poles. The system does not handle such configurations. Sorry - try something else.

2) "TOO MANY CURVES"

No more than ten curves may be displayed at one time on the transient response display. This number has been exceeded. Clear the display and try again. If a curve has been erased with the light pen (see Section 4.8), it is not displayed, although the graphic orders to create this curve do remain in the buffer. Therefore, such erased curves must be considered as part of the total number of displayed curves until the display is cleared.

3) "COMPUTATION INTERRUPT"

An interrupt occurred in computing the transient response curve. This is possible when the current pole-zero map is complicated, the multiplicity of a pole is greater than three or the range of the time interval is large with poles in the right half-plane. Decrease the upper bound of the time interval and try again. If this fails to correct the problem, possibly the response may not be obtainable.

4) "PLOT ERROR"

A floating point error occurred in plotting the curves. Probably a floating point value is too large for the plotting subroutines. Decrease the upper bound of the time interval and try again.

5) "INVALID FUNCTION KEY"

The last function key used has no meaning with the current display.

6) "INVALID DETECTION"

The light pen has detected an invalid part of the transient response display. Only a response curve may be detected. This error will occur only after PFK 14 has been depressed (see Section 4.8).

4.8 Special Features

There are other features of the system to facilitate use.

Clearing a Display

PFK 9 (CLEAR) applies to either the pole-zero map display or the transient response display, whichever is current. From the pole-zero map display, the effect is to remove all poles and zeros from the display. The constant multiplier is unaffected. If both input and system poles and zeros are on the display, the first depression of PFK 9 will remove only the inputs. From the transient response display, all response curves are removed.

Erasing One Response Curve

To erase a single curve from the transient response display, first detect any part of that curve with the light pen and then depress PFK 14 (ERASE). A detection is successful only if the alarm sounds. Detection of any part of the display other than a response curve results in the error message "INVALID DETECTION" in the message area when PFK 14 is depressed. Note that the curve is actually blanked - not erased - and, therefore, considered by the system as a displayed curve.

Built-in Inputs

Certain pole configurations are commonly used in pole-zero map analysis. PFK's are available to generate automatically some of these structures on the pole-zero map display. These are:

PFK 11 - STEP	Displays a single pole at the origin of the pole-zero map.
PFK 12 - RAMP	Displays a double pole at the origin of the pole-zero map.
PFK 13 - SIN	Displays poles at $\pm j$ of the pole-zero map.

These may be used from system or input mode.

Cursor Positioning

PFK 16 (PLACE) may be used with the grid parameter display or the pole-zero map display to re-position the cursor. Use with the pole-zero display places the cursor to the first position of the data pad area as in Figure 4-4. Use from the grid parameter display alternately sets the cursor to the first field of the pole-zero grid fields or the transient response grid fields depending on the current cursor position.

Recovery

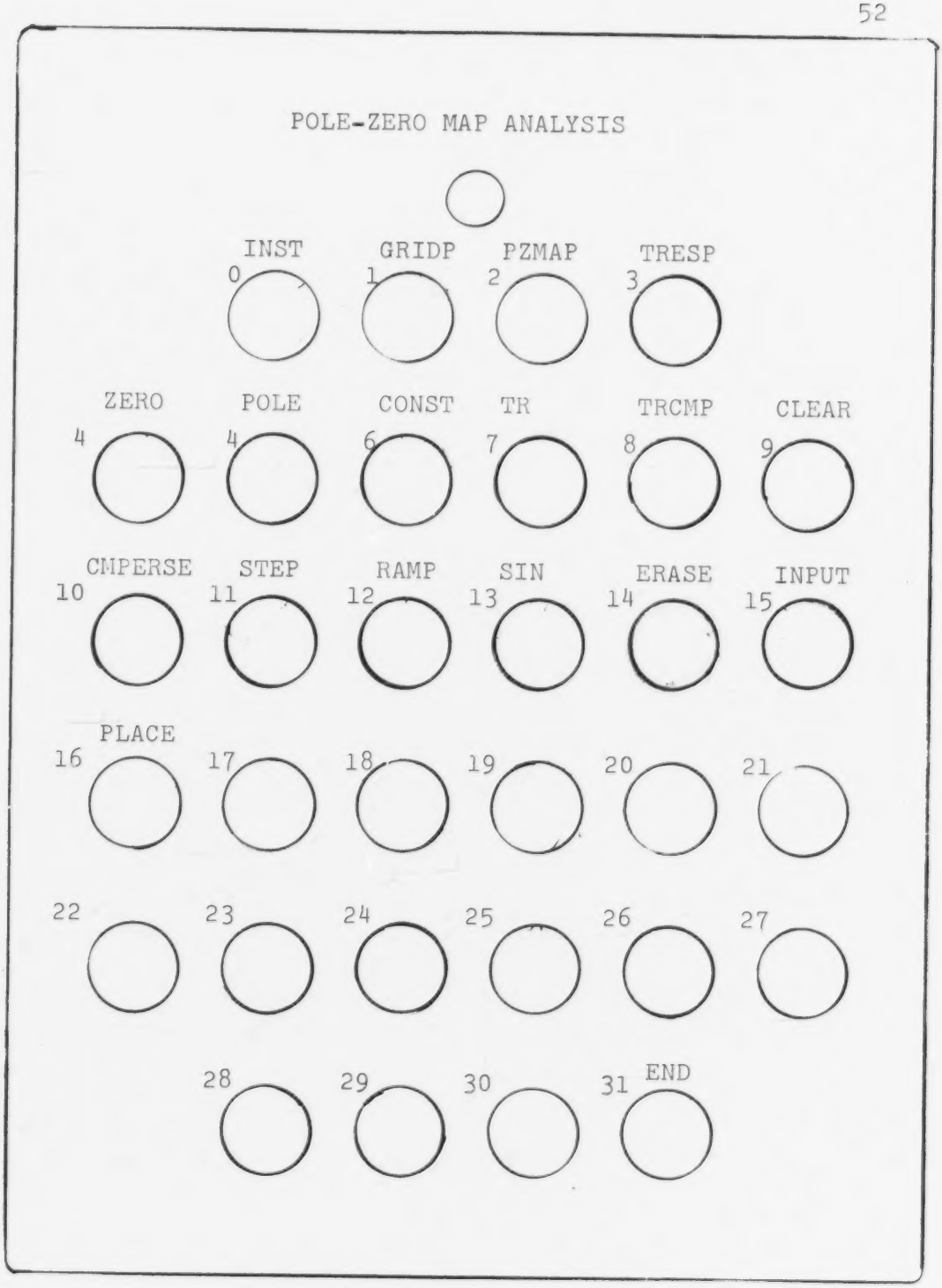
PFK 0 (INST) may be used at any time during system operation. This displays the general instructions and resets the system to its initial state to allow the user to start anew.

4.9 System Termination

When through using the system, depress PFK 31 (END). This should blank the screen and turn off all PFK indicator lights.

4.10 Programmed Function Keys

The following is a detailed description of the active programmed function keys used by the system. The label is the mnemonic identifying the key on the plastic overlay for the programmed function keyboard (see Figure 4-6).



Programmed Function Keyboard

Plastic Overlay

Figure 4-6

PFK #	LABEL	FUNCTION
00	INST	The general instructions are displayed. This may be used at any time. All other displays are reset to their initial state.
01	GRIDP	The grid parameter fields are displayed for the pole-zero grid and the transient response grids. The cursor is positioned to the first pole-zero field.
02	PSMAP	The pole-zero grid and the poles, zeros and constant multiplier of the current system transfer function are displayed. System mode is initiated.
03	TRESP	The transient response grid and any response curves computed previously are displayed. No response computation is invoked.
04	ZERO	Zeros are displayed on the pole-zero grid at the coordinates specified in the data pad.
05	POLE	Poles are displayed on the pole-zero grid at the coordinates specified in the data pad.
06	CONST	The constant multiplier is displayed on the pole-zero grid according to the contents of the data pad. If in input mode, the new multiplier is the product of the

current multiplier and data pad multiplier. If in system mode, the contents of the data pad are copied into the constant multiplier field.

07	TR	The transient response grid display and any previous curves computed are displayed. The transient response of the current pole-zero configuration is then computed and displayed.
08	TRCMP	The transient response grid and any previous curves computed are displayed. The transient response of the current pole-zero configuration and its components are then computed and displayed.
09	CLEAR	The pole-zero grid is cleared of all poles and zeros, or the transient response curves depending upon whichever display is current.
10	CMPERSE	The component curves resulting from the last response computation are erased. If the last response computation was instigated by PFK 7, no curves are erased.
11	STEP	A single pole is displayed at the origin of the pole-zero grid.

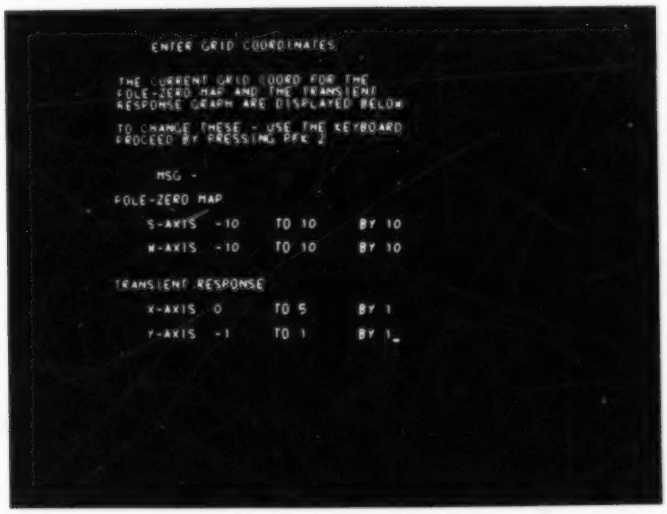
12	RAMP	A double pole is displayed at the origin of the pole-zero grid.
13	SIN	A pair of complex poles are displayed at $\pm j$ of the pole-zero grid.
14	ERASE	The response curve detected by the light pen in the previous step is erased.
15	INPUT	The input mode is initiated. The message "NOW IN INPUT MODE" is displayed in the pole-zero message area and the alarm sounds. Henceforth, all inputs will be associated with the input function until PFK 2 is again used.
16	PLACE	The cursor is positioned according to the current display. If the pole-zero grid display is current, the cursor is set to the first position of the data pad. If the grid parameter display is current, the cursor is set to either the first field of the pole-zero grid fields or the transient response fields depending upon the current cursor position.
17-30		Inactive. Initiate no response.
31	END	Terminate the system.

4.11 Examples

The examples included in this section are for the user being introduced to the system for the first time. They are designed to familiarize the user with the basic operation of the system. Most of the facilities of the system are illustrated at some point within these examples. The figures are actual photographs of the 2250 display area taken with a Polaroid camera. The σ and $j\omega$ axes have not been included in the pole-zero map pictures so that the pole/zero positions may be more easily seen.

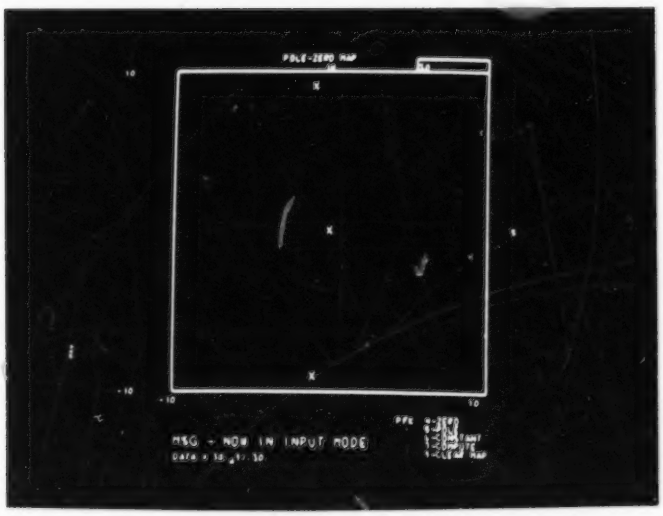
Example 1 takes the user through the process of setting the grid parameters. Example 2 illustrates the difference between requesting the system response alone and requesting the system response with its components. Example 3 illustrates multiple poles, input functions and superposition of responses. The examples are presented as sequences of chronological user steps with related comments. A reference to a figure within the sequence indicates the status of the display at the completion of that particular step.

Example 1: Set the grid parameters for the transient response grid as follows: x ranges from 0 to 5 in increments of 1. y ranges from -1 to 1 in increments of 1 (see Figure 4-7-A).


ENTER GRID COORDINATES
THE CURRENT GRID COORD FOR THE
POLE-ZERO MAP AND THE TRANSIENT
RESPONSE GRAPH ARE DISPLAYED BELOW
TO CHANGE THESE - USE THE KEYBOARD
PROCEED BY PRESSING PFK 2
MSG -
POLE-ZERO MAP
S-AXIS -10 TO 10 BY 10
W-AXIS -10 TO 10 BY 10
TRANSIENT RESPONSE
X-AXIS 0 TO 5 BY 1
Y-AXIS -1 TO 1 BY 1

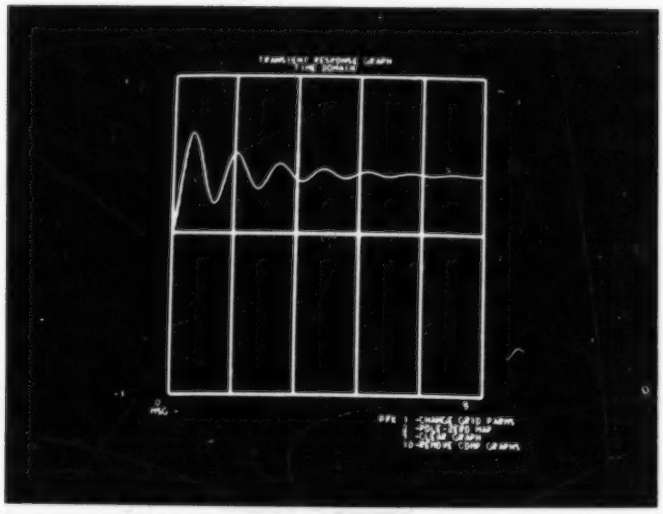
Grid Parameter Display

Figure 4-7-A



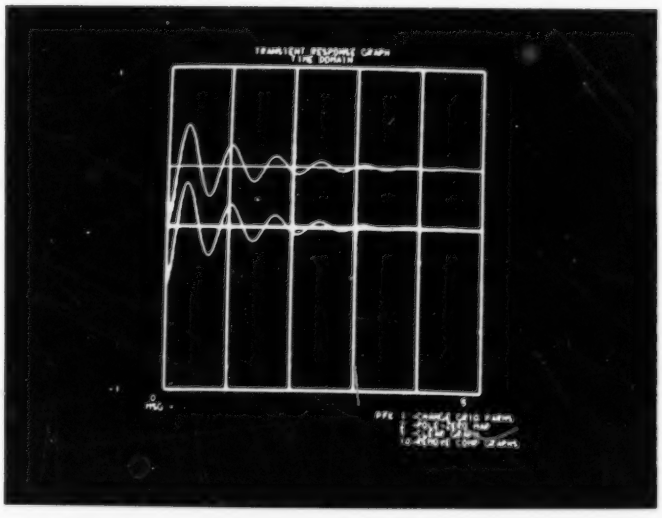
Pole-Zero Maps With Inputs

Figure 4-7-B



Transient Response

Figure 4-7-C



Transient Response with Component Curves

Figure 4-7-D

Action		Comments
1.	GRIDP	PFK 1-grid parameter display
2.	PLACE	PFK 16-cursor to first TR field
3.	Key-in "0"	Lower x value
4.	jump key	Cursor to upper x field
5:	Key-in "5"	Upper x value
6.	jum key	Cursor to x increment field
7.	Key-in "1"	x increment value
8.	jump key	Cursor to lower y field
9.	Key-in "-1"	Lower y value
10.	jump key	Cursor to upper y field
11.	Key-in "1"	Upper y value
12.	jump key	Cursor to y increment field
13.	Key-in "1"	y increment value

When complete, the grid parameter display should appear as in Figure 4-7-A. The transient response grid would appear as in Figure 4-7-C.

Example 2: The grid parameters are set as in Figure 4-7-A. Display a pair of complex poles at $(-1,\pm 9j)$ with constant multiplier = 30. First display the response to a unit step without the components and then display the response to a unit step with the components.

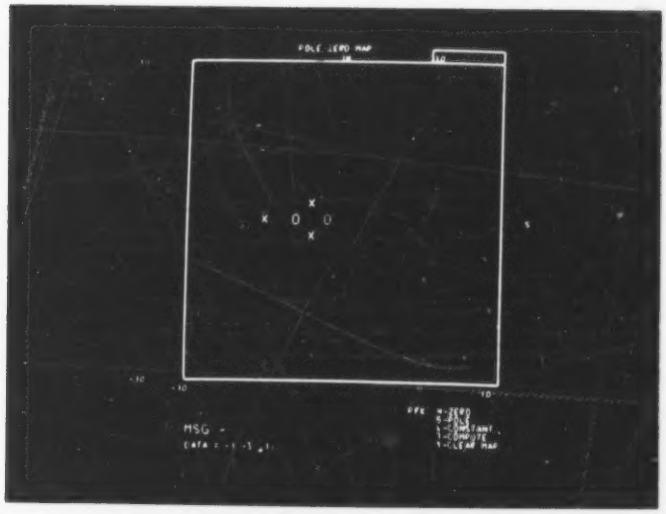
Action		Comments
1.	PZMAP	PFK 2-Display pole-zero grid
2.	CLEAR	PFK 9-Remove any poles/zeros
3.	DATA = 30;	Enter constant multiplier
4.	CONST	PFK 6-Display constant multiplier
5.	PLACE	PFK 16-Cursor to first data pad position
6.	DATA = (-1,9);	Enter pole coordinates
7.	POLE	PFK 5-Display poles
8.	INPUT	PFK 15-Begin input mode
9.	STEP	PFK 11-Display pole at origin (Figure 4-7-B)
10.	TR	PFK 7-Display response curve (Figure 4-7-C)
11.	CLEAR	PFK 9-Remove the response curve
12.	PZMAP	PFK 2-Display pole-zero map
13.	Repeat steps 8&9	Re-display the input function
14.	TRCMP	PFK 8-Display response and components (Figure 4-7-D)
15.	CMPERSE	PFK 10-Erase component curves (Figure 4-7-C)

Example 3: Set the transient response grid parameters so that x ranges from 0 to 10 by 2 and y ranges from -5 to 5 by 5. Display a system with poles at -5 and (-2,±1j), zeros at -1 and -3 and constant multiplier = 10 Examine the response to a step input with

magnitude = 2.5. Add another pole at -5 and repeat for the same input. Look at the response of this system to an input of the form 2.5sin(2x) (pole at (0,±2j), constant multiplier = 2.5).

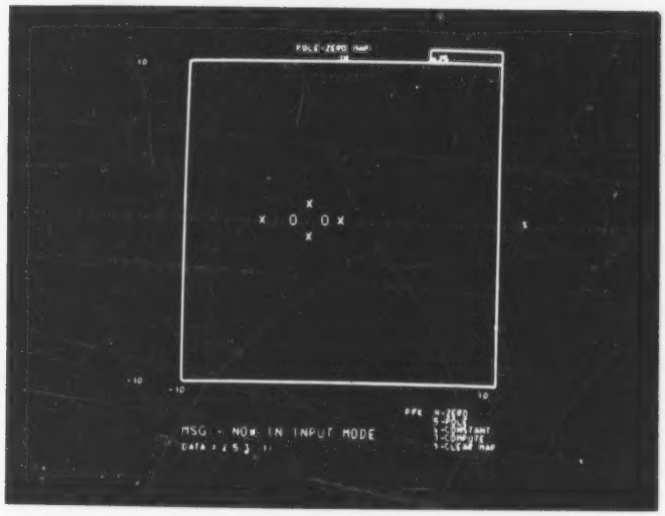
Action		Comments
1.	Set grid parameters	Follow example 1 procedures
2.	PZMAP	PFK 2-display pole-zero map
3.	DATA = -5,(-2,1);	Enter pole coordinates
4.	POLE	PFK 5-display poles
5.	PLACE	PFK 16-reset cursor
6.	DATA = -1,-3;	Enter zero coordinates
7.	ZERO	PFK 4-display zeros
8.	PLACE	Reset cursor
9.	DATA = 10;	Enter system constant multiplier
10.	CONST	PFK 6-display constant multiplier (Figure 4-8-A)
11.	INPUT	PFK 15-begin input mode
12.	STEP	PFK 11-display pole at origin
14.	DATA = 2.5;	Enter input constant multiplier
15.	CONST	Display new constant (Figure 4-8-B)
16.	TR	PFK 7-Display system response (Figure 4-8-C)

Action		Comments
17.	PZMAP	Display pole-zero map (Figure 4-8-A)
18.	DATA = -5;	Enter pole coordinates
19.	POLE	Display pole
20.	Repeat steps 11-15	Re-display the input function (Figure 4-8-D)
21.	TR	Display system response (Figure 4-8-E)
22.	PZMAP	Display pole-zero map
23.	INPUT	Begin input mode
24.	DATA = (0,2);	Enter input pole coordinates for sin
25.	POLE	Display input poles
26.	Repeat steps 13-15	Input constant multiplier (Figure 4-8-F)
27.	TR	Display system response (Figure (Figure 4-8-G)



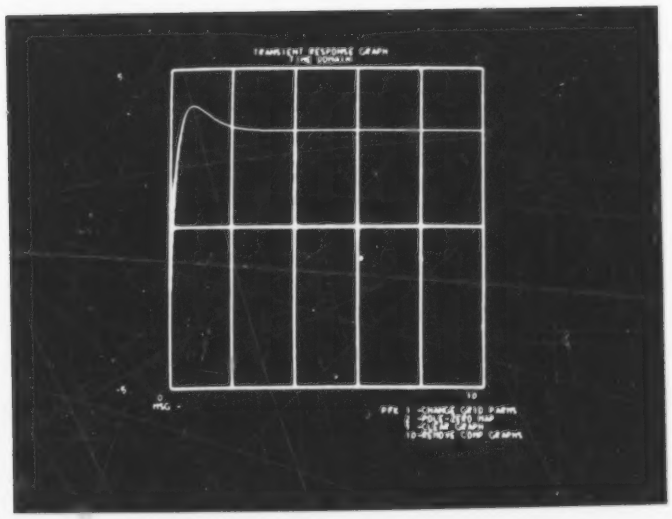
System Map

Figure 4-8-A



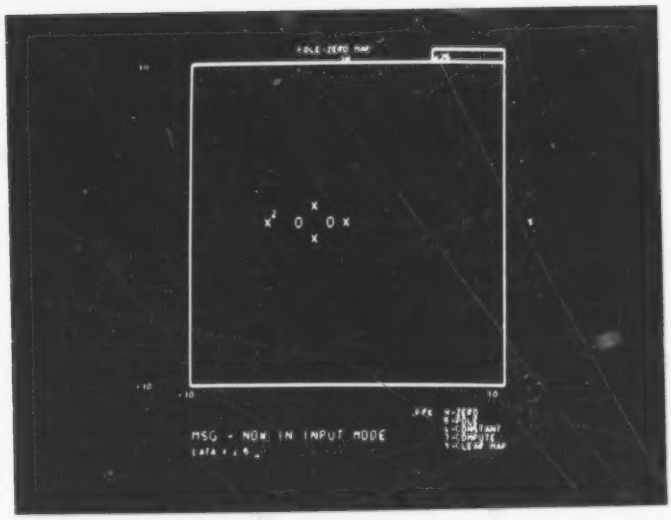
System Map with Inputs
(Note constant multiplier)

Figure 4-8-B



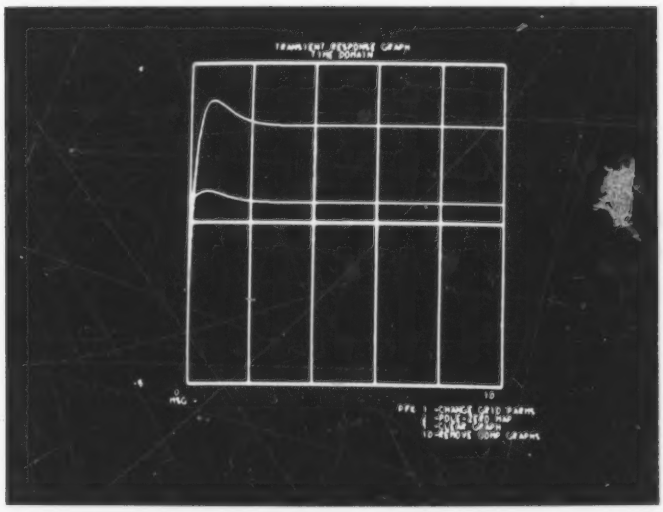
Transient Response to 4-8-B

Figure 4-8-C



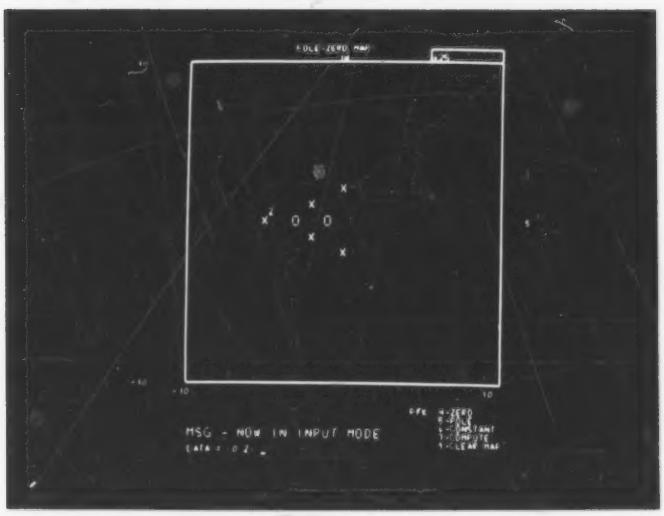
Multiple Pole at (-5,0)

Figure 4-8-D



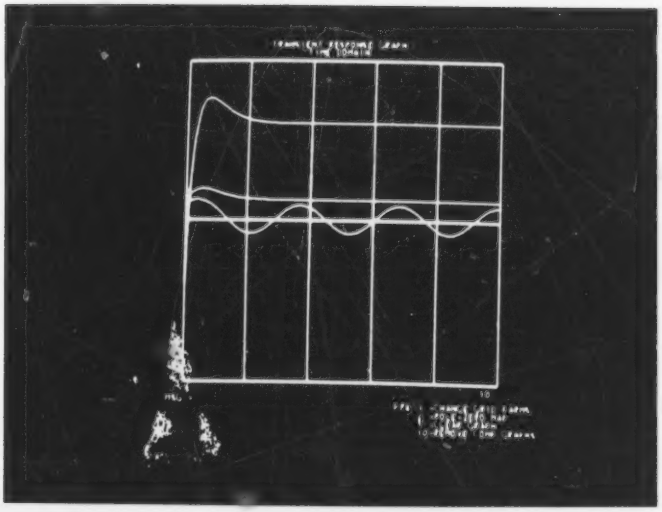
Response of 4-8-D
Superposed with 4-8-C

Figure 4-8-E



Sin input.

Figure 4-8-F



Response of 4-8-F
Superposed with 4-8-E

Figure 4-8-G

Chapter 5

SOFTWARE DESCRIPTION

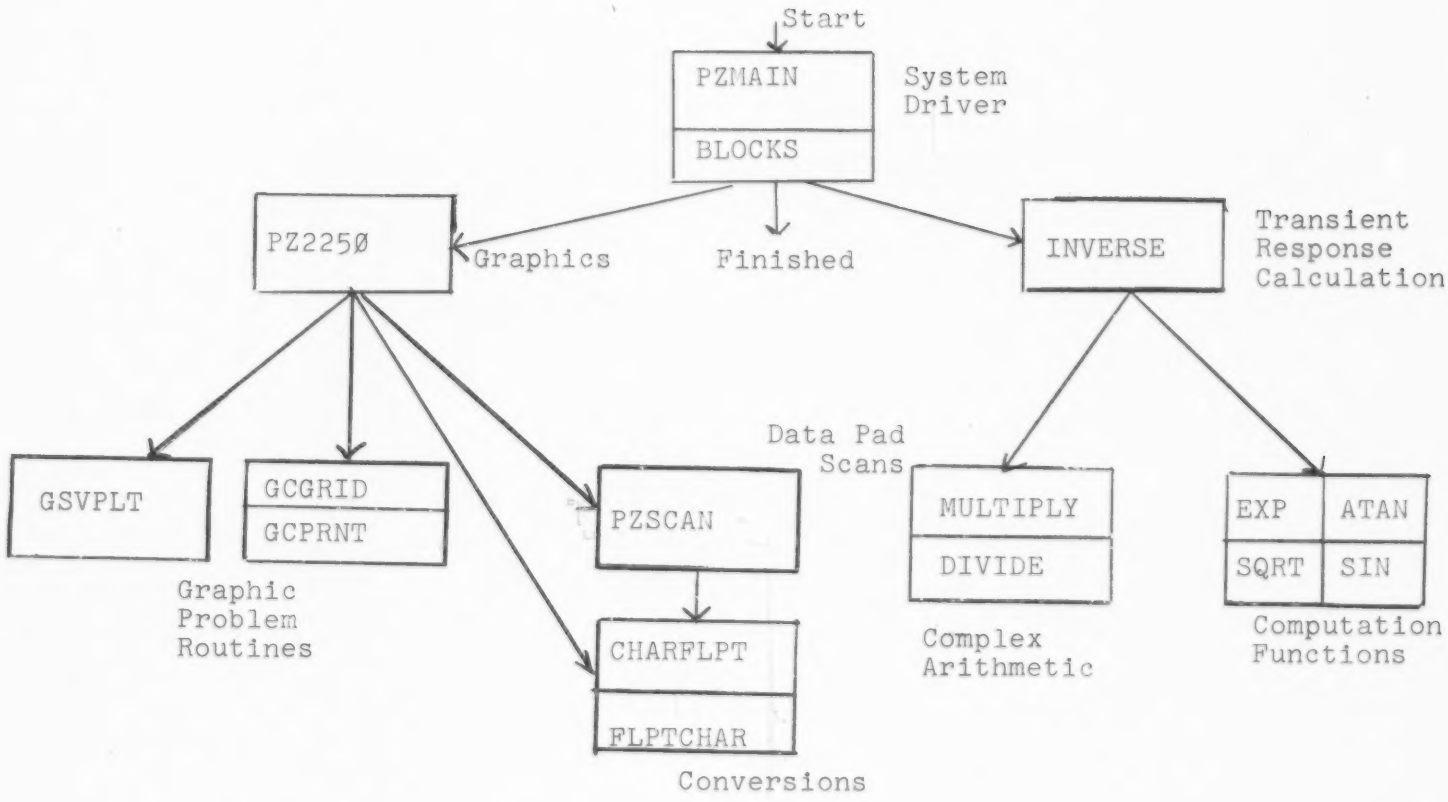
This chapter describes the internal structure and characteristics of PZMAP. All related software was developed in Operating System/360 Assembler Language running under Operating System/360 on an IBM System/360 Model 40. Data display and man-machine communication with the 2250 display device are accomplished with graphic macros and problem oriented subroutines provided by IBM [11].

5.1 System Components

The system consists of 16 modules. Their names and relations to each other are shown in Figure 5-1. Program execution begins and terminates in PZMAIN. Calls from one routine to another follow the directions of the arrows in the figure and subsequent returns of control follow the arrows in the opposite direction. The complete computer listings for the routines are located in Appendix F. A brief discussion of each routine follows.

PZMAIN

PZMAIN is the executive module of the system. It



System Components

Figure 5-2

performs initial and terminal system linkage and monitors all communications between PZ2250 and INVERSE. All common data storage areas are defined within PZMAIN and it resides in core at all times. The calling structures are arranged so that a normal return from PZ2250 results in an immediate call to INVERSE and the return from INVERSE results in another call to PZ2250 at the secondary entry point TRPLOT. The system is terminated by a return from PZ2250 to the entry point LEXIT within PZMAIN.

BLOCKS

BLOCKS contains the graphic data control block (GDCB) and the graphic attention control block (GACB).

PZ2250

PZMAIN passes control to PZ2250 (the graphics routine) after system initialization is complete. PZ2250 does the graphic initialization and handles all communication with the 2250 display device. This includes processing all attention interrupts and creating all graphic displays.

Initially, the general instructions display is generated upon the CRT and the program enters a wait state. When a user attention is recognized, program execution begins again. The attention is analyzed and a branch is effected to the appropriate program segment. When that execution is completed the program again

enters a wait state.

To compute the transient response curve, PZ2250 returns control to PZMAIN which in turn calls INVERSE for response computation. When completed, PZ2250 is once again called by PZMAIN at the secondary entry point TRPLOT ready to display the response.

INVERSE

INVERSE is called by PZMAIN to compute the points representing the inverse Laplace transform curve (transient response) of the current pole-zero configuration over a given time interval (XL to XH). This routine is structured with a driver segment and seven internal subroutines. It effects the algorithm discussed in Section 2.6. Data are obtained by external reference to data structures in PZMAIN.

PZSCAN

PZSCAN is called by PZ2250 to scan an input character string obtained from the data pad of the pole-zero map display. The scaled grid coordinates for poles or zeros are plucked from the string according to a set of syntax rules. Associated tables and pointers are also updated in the process.

Complex Arithmetic Routines

The routines MULTIPLY and DIVIDE are both called from INVERSE to perform complex multiplication and division respectively. The program logic was obtained from the IBM modules IHEMZW and IHEDZW respectively [8].

Computational Routines

The routines EXP, ATAN, SQRT and SIN are all called by INVERSE to perform specific computational functions. The purpose of each with a given argument x is:

EXP - computes e^x

ATAN - computes arc tangent of x

SQRT - computes square root of x

SIN - computes sin(x)

These routines were written by Dr. Alfred Oliver and originally used in a system for teaching numerical analysis [12].

Problem Oriented Routines

These routines GCGRID and GCPRNT are provided by IBM in the graphic programming services for the 2250 device. GCGRID is used to generate Cartesian grid displays and GCPRNT is used to generate alphanumeric characters on the CRT. These routines reside in LINKLIB (an operating system library). Both are called by PZ2250.

GSVPLT

This routine is a corrected version of the IBM problem oriented routine of the same name. GSVPLT is a plotting routine used to plot characters within a grid (poles and zeros) and curves (transient responses). It is called by PZ2250.

Conversion Routines

CHARFLPT converts an EBCDIC character string to a floating point representation and FLPTCHAR converts a floating point form to an EBCDIC character string. CHARFLPT is called by both PZ2250 and PZSCAN, and FLPTCHAR is called by PZ2250 only. These routines were also written by Dr. Alfred Oliver.

5.2 Important Data Structures

All data used by both PZ2250 and INVERSE are defined in PZMAIN and referenced as external variables. This principally includes all data structures required to represent the pole-zero configuration and the response curves.

The pole-zero configuration is represented by a set of similar structures for the poles and zeros. These are:

UPTAB,VPTAB - Contiguous 10-word tables for the real and imaginary coordinates of a pole position.

PMULT - A 20-byte table for the multiplicities of the corresponding pole positions specified in UPTAB. A 2-byte field exists for each pole. The first byte contains the total multiplicity of the pole. The second byte contains the number of poles of the total multiplicity which are associated with the input function.

CPSA - Pointer to the next available address in UPTAB.

UZTAB,VZTAB - Similar to UPTAB and VPTAB for the zero positions.

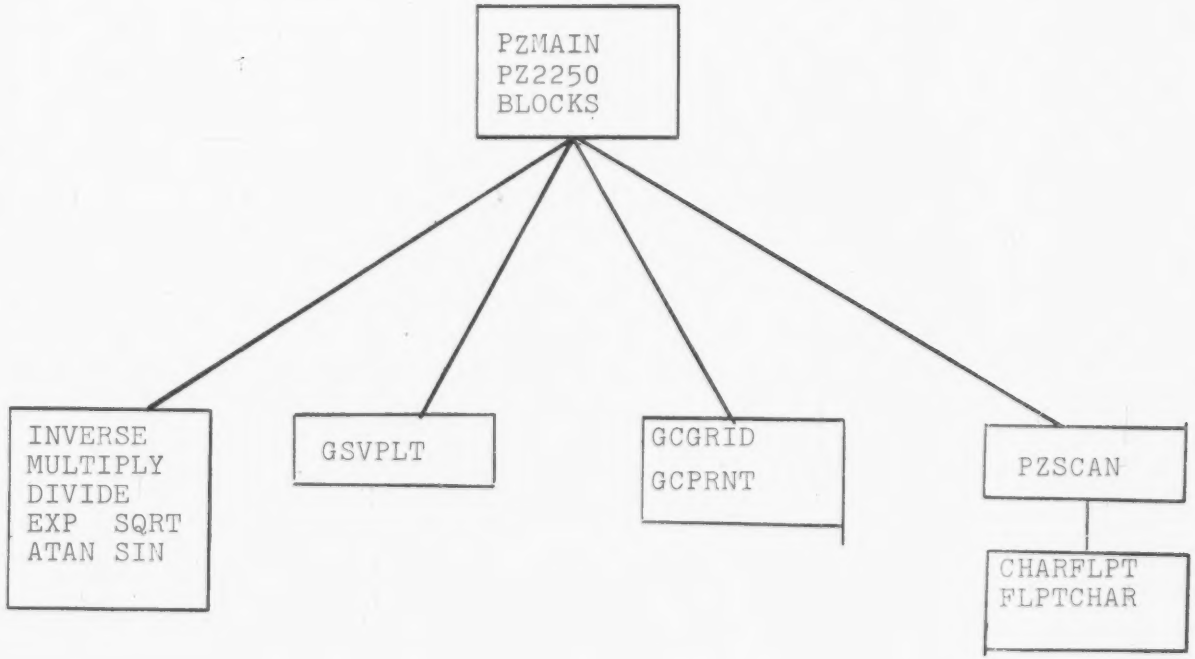
ZMULT - Similar to PMULT for the zero multiplicities.

CZSA - Similar to CPSA.

The transient response plot information is maintained in ten 180-word plot tables, TAB1, TAB2,···,TAB10. PLOTPTR always points to the next available table.

5.3 Core Requirements

PZMAP was developed in a 44K byte partition of main memory. The overlay structure utilized is shown in Figure 5-2. All modules within a given block are loaded into memory as one unit. While a module is in use, all modules in a higher block are in main memory. With this overlay structure, approximately 27K bytes of memory are required for PZMAP programs. The operating



PZMAP Overlay Structure

Figure 5-2

system allocates additional storage during execution making the maximum requirement about 39K bytes. This could be reduced by implementing the system modifications described in Section 5.5.

5.4 Implementation

Ideally there should be few problems in implementing PZMAP in any IBM System/360 environment with an IBM 2250 Model 1 attached. All programs were developed in assembler language and should be independent of the operating system in use (release 16 was used in program development). The graphic programs were designed for use with an 8K byte buffer storage. Alterations for use with 2250's with a 4K buffer storage would require only minor program changes in terminal buffer addresses used in PZ2250. However, this would allow the display of only two response curves at a time. Modifications for a 2250 without a buffer would be significant.

The least-used attention mechanism is the light pen (used only to detect a response curve for erasure) and the system could easily be adapted to a device with no light pen. However, considerable system re-design would be necessary for use with a device without programmed function keyboard and/or manual input facilities.

To invoke the system the following job control releases the PZMAP load module from a disk data set into the graphics partition and initiates program execution in PZMAIN.

```
//PZMAP     JOB (UNC.IS.E5103,,T),'BEYER,W',MSGLEVEL=1,   *
//              CLASS=G,TYPRUN=HOLD
//              EXEC PGM=STEPLIB
//STEPLIB   DD  DSNAME=UNC.IS.F7027.ISDEPT.GRAPHICS(PZMAP)
//DISPLAY   DD  UNIT=DISPLAY
//SYSUDUMP  DD  SYSOUT=A,SPACE=(TRK,(40,20),RLSE,CONTIG)
```

5.5 Hind-Sight

Presently PZ2250 is a large module (16K) containing all of the graphic functions in the system. An initial step in system revision should be a complete modularization of this routine into about 10 smaller routines reflecting the present internal structure of PZ2250. This would make future extentions easier and more efficient and allow for a tighter overlay structure (about 18K bytes for PZMAP programs).

To accomplish this, however, a problem arises because of shortcomings in the IBM-provided graphic macros (e.g., GWRITE, GREAD,GCNTRL). These macros include absolute and relocatable references to the graphic data control block (DCB). This creates difficulties in assembling the macros in a subroutine which references a graphic DCB external to the subroutine.

An external DCB is desirable for several reasons. The definition of the 32-byte DCB would be the same in each subroutine and, therefore, an external reference

saves space and eliminates redundancy within the subroutines. Also, in an overlaying program environment the DCB should remain in main memory at all times. In this respect, a structure with separate graphic DCB's for each graphic subroutine eliminates the possibility of overlaying these subroutines.

To circumvent this problem a separate control section, containing only the graphic DCB (and the graphic attention control block), should be included in the assembly of each graphic subroutine (for an illustration refer to the BLOCKS control section at the beginning of the PZ2250 assembly). This will allow a proper assembly of the graphic macros. If these DCB control sections have the same name in each graphic assembly, a reference to that name in the link edit step will link only the first reference to that name to the load module. Therefore, only one copy of the graphic DCB is included in the load module and each graphic subroutine references the same DCB.

PZ2250 is currently not serially reusable because of several data items included within it. In particular these are the grid parameters (both floating point and character representations) for both grid displays and the buffer address tables with related pointers. The effect is that PZ2250 must reside in core at all times. Otherwise, these data items will be re-initialized by the operating system each time PZ2250 is dynamically reloaded. This may be corrected by moving the data into PZMAIN and altering the references in PZ2250 to these data.

Chapter 6

CONCLUSIONS

6.1 Evaluation of the System

Accuracy

The system effectively computes the transient responses for pole-zero configurations of varying complexity. The range of complexity varies with the number, relative positions, and magnitudes of poles and zeros and the time interval over which the response curve is displayed. Accurate responses have been obtained for configurations with up to eight and nine distinct poles. This is certainly a reasonable upper bound in an educational environment.

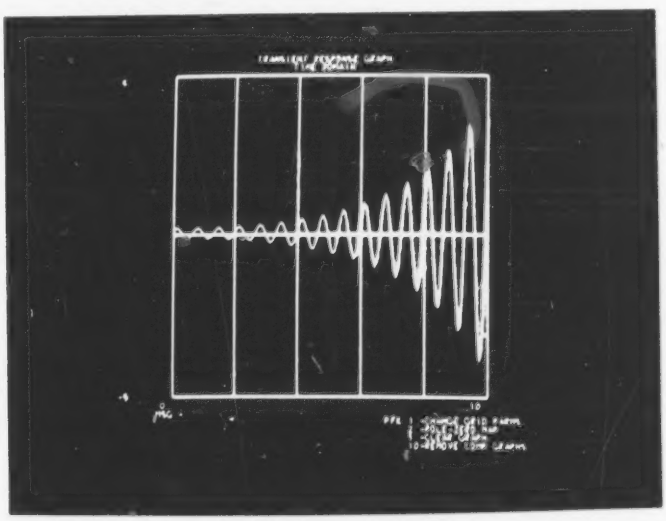
The responses for systems with pole multiplicities greater than three may not be computed with the current offset ϵ value of .011 (see Section 2.6). Pole multiplicities of three result in responses with considerable error in the magnitude though the shape of the curve is not distorted. A reasonable statement is that pole multiplicities of two are accurately displayed while any multiplicities greater than two are at best risky. Zero multiplicities greater than one are not a direct problem as they affect only the magnitudes of the residues.

The ability to compute responses of systems with poles of multiplicity $h > 1$ is largely a function of the offset ϵ. The partial fraction expansion of these functions results in an $\epsilon^{(h-1)}$ term in the denominator of the residues of the offset poles. Therefore, for small ϵ, round-off error becomes a dominant consideration.

The effect of offsetting poles of multiplicity two with the current ϵ is normally not obvious over reasonably small intervals of time (less than 50). A conspicuous exception is the ramp function (double pole at the origin, $y = x$ when constant multiplier equals 1). The ramp function displayed over a time interval of 0 to 10 passes through the point (9.6,10). This is disconcerting to the user who naturally expects the line to pass directly through the point (10,10). For larger time intervals the error becomes increasingly more severe because a pole has been offset into the right half-plane.

A very desirable facility would allow the user to vary the offset ϵ by manual input from the alphanumeric keyboard. This would give the user some control over the precision of the response curves as well as the range of allowable pole-zero configurations. For example, if the user desires to examine a system with a large multiplicity, ϵ could be increased. For a very accurate look at a response to a system with a double pole, ϵ could be made smaller.

Poles in the right half-plane contribute rising exponentials to the system response. These get large very fast. Such pole-zero configurations must have a small upper bound for the time domain (probably less than 10). This is adequate in most cases to illustrate the responses of unstable systems. Figure 6-1 is a photograph of a sinsusoidal response increasing within an exponential envelope over a time interval of 0 to 10.



Example of Unstable Response

Figure 6-1

Speed

The response time is excellent. The wait between pressing the compute function key and the display of the transient response curve varies from instantaneous to about 8 seconds. This depends principally upon the

complexity of the pole-zero configurations. The response of a simple system is displayed immediately, whereas the response of a complicated system with multiple poles takes longer.

The response time is also a function of the current computing load of the IBM System/360 Model 40. During peak computing periods the overall response of the system is somewhat slower. This is despite the fact that the graphic partition of main memory has a high priority for the CPU.

Effectiveness

The system has been informally tested with several graduate students who have all had the experience of at least one prior course in automatic control systems. From these tests, and my own extensive testing of the system, certain qualitative judgments may be made about the effectiveness of the system. No attempt was made to make quantitative measures of the system utility. Some work in the area of justifying the use of interactive displays in teaching environments has been done by Dr. Alfred Oliver [12].

In the classroom. The system is extremely powerful when used by a person well familiar with the subject matter. In a short period of time a wide spectrum of control system concepts may be illustrated from a variety of viewpoints. The infinite number of possible pole-zero

configurations and the infinite number of possible time intervals allow a great flexibility in coping with spontaneous classroom questions. A particularly useful feature in this respect is the ability to superpose one response curve over another curve. This is instructive for examining the effect upon the response of moving and/or adding poles and zeros to the current configuration. The system utility appears to be limited only by the imagination of the teacher.

There are problems in using the system in the classroom. While the 2250 display device is on, the noise of the cooling fans is sufficient to muffle somewhat the voice of the teacher. This is not a problem if the teacher speaks loudly and clearly. It is not possible to turn the device on and off as needed because a ten minute warm-up period is necessary each time the device is on.

In order to enter data from the alphanumeric keyboard, the teacher must bend over slightly and turn either his back or side (depending upon his agility) to the students. This could be time consuming if the teacher is 1) unprepared, 2) unfamiliar with using the system, or 3) ill-coordinated for such activities. Therefore, the teacher must make an initial investment of time to learn how to operate the system smoothly, as well as checking out relevant examples prior to

each class session. The question of whether or not a teacher will do this has not been investigated. An alternative approach would be to use photographs or slides of the display area prepared previously.

6.2 Criticisms

Two shortcomings of the system should be mentioned. The first of these concerns the level of error messages and user instructions. A desirable characteristic of the system would be that a new user, who is familiar with 2250 operation, could operate the system without prior reference ot the user manual. This is currently not the case. More detailed error messages and instructions as well as "rescue" instructions could rectify this.

The second criticism concerns the lack of edit facilities with the pole-zero display. Currently, the only edit facility is the CLEAR function key which removes all poles and zeros from the display. This means that to move or delete a pole or zero from the map the user must clear the map and completely reconstruct the desired map. Desirable facilities would include:

1) light pen detect for removal of a pole/zero.
2) Deletion of the last pole/zerō entry via a function key.

3) Deletion of all poles or all zeros via a function key.

6.3 Extensions

There are desirable and logical extensions to the system which could be implemented. Some of these are listed below:

1) Translator facilities for pole/zero function specification in algebraic form.
2) Retrieval facilities which would allow an instructor to create displays, tag them, and then save them on auxiliary storage for recall during classroom lectures.
3) Inquiry facilities which would give the user access, for example, to the coordinates of displayed poles/zeros or their respective residues, or time constants, frequencies, or phase shifts of response curves.
4) An ability to vary the offset ϵ discussed in Chapter 2 and Section 6.1.

Appendix A

TABLE OF LAPLACE TRANSFORMS [4]

#	F(s)	f(t)	Comments
1	$\frac{1}{s}$	$u(t)$	Unit step at t - 0
2	$\frac{1}{s^2}$	t	Unit ramp starting at t = 0
3	$\frac{n!}{s^{n+1}}$	t^n	(n = 1, 2, ···)
4	$\frac{1}{s + b}$	e^{-bt}	
5	$\frac{1}{(s + b)^2}$	te^{-bt}	
6	$\frac{\omega}{s^2 + \omega^2}$	sin ωt	
7	$\frac{s}{s^2 + \omega^2}$	cos ωt	
8	$\frac{s}{s+b+j\omega_0} + \frac{\overline{K}}{s+b-j\omega_0}$	$2\|K\|e^{-bt} \sin(\omega_0 t + \emptyset)$	

where $K = |K| \angle < K$

$= n + jm$

$|K| = \sqrt{n^2 + m^2}$

$\angle K = \tan^{-1}(\frac{m}{n})$

$$\overline{K} = n - jm$$

$$\phi = \tan^{-1}\left(\frac{n}{m}\right)$$

9	$\frac{1}{s(s + b)}$	$\frac{1}{b}\left(1 - e^{-bt}\right)$
10	$\frac{s + a}{s(s + b)}$	$\frac{a}{b} + \left(1 - \frac{a}{b}\right)e^{-bt}$

Definition of Inverse Laplace Transform [4]

The inverse Laplace transform, f(t), of F(s) where F(s) is the Laplace transform of f(t) is defined as follows:

$$f(t) = \frac{1}{2\pi j}\int_{c-j\infty}^{c+j\infty} F(s)ds$$

where c is a constant which must not exceed a certain value determined by F(s).

Appendix B

SAMPLE EXERCISES

The following sample exercises illustrate some of the types of problems for which the system is applicable. It is assumed that the student is familiar with operating the system and it is the responsibility of the student to adjust the grid parameters accordingly for each probalem.

1. Given the electrical circuit in Figure B-1 with R_1 = 5 ohms, R_2 = 10 ohms, C = 50 μfarads and E = 5 volts. Using $e_1(t)$ as the input, perform a Laplace

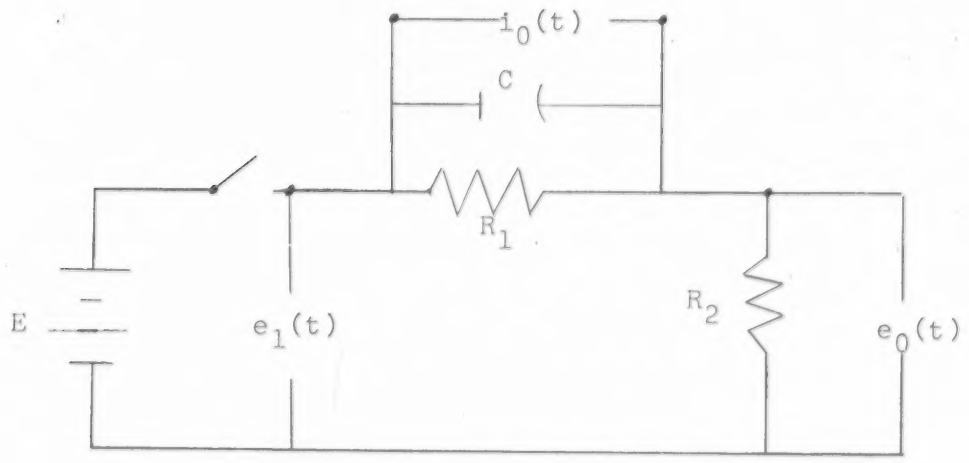


Figure B-1

transform analysis of the system response with both $i_0(t)$ and $e_0(t)$ as the output. Sketch these responses. Using the results of your computations, display the pole-zero configurations for the systems and display the transient responses. Compare your responses with the displayed responses. Investigate the effect upon the responses by increasing and decreasing R_1, R_2 and C independently.

2. You are given a system with poles at $(-1, \pm 9j)$. Examine the response to a step input. Normalize the response with respect to the final value of the response. What is an estimate of the damping ratio? What is the effect of decreasing the $j\omega$ coordinate? What is the effect of increasing the σ coordinate?

3. A pole-zero system has poles at -2, -3, -20. Consider the response to a step input. Which pole has the least effect upon the system response? Why? Which has the most significant effect? Why? What is the sign of the residues associated with each pole? The introduction of a pole at -5.1 and a zero at -4.9 has what effect upon the response? Add a pole at +.2. What happens to the response? Increasing the size of this pole does what to the response?

4. A system has poles at -1 and $(-3,\pm 3j)$. Examine the response to a step input. Examine the response to a pure sinusoidal input $(0,\pm j)$. Compare the steady-state responses for the two inputs. Are both systems stable? How will the latter response differ from the system response with an input with poles at $(0,\pm 9j)$.

Appendix C

LIST OF PROGRAMMED FUNCTION KEYS

AND THEIR FUNCTION

Overlay 0

#	Key	Function
0	INST	Display the general instructions.
1	GRIDP	Display the fields to set new grid parameters for the pole-zero map grid and the transient response grid.
2	PZMAP	Display the current pole-zero map grid and any current ples and zeros.
3	TRESP	Display the current transient response grid and any current response curves.
4	ZERO	Display zeros at the positions specified in the data pad.
5	POLE	Display poles at the positions specified in the data pad.
6	CONST	Display new constant multiplier using the entry in the data pad.
7	TR	Compute and display the transient response of the current pole-zero map.
8	TRCMP	Compute and display the transient response and the components of the transient response of the current pole-zero map.
9	CLEAR	Clear the current display.
10	CMPERSE	Erase the component curves from the last component computation (PFK 8).
11	STEP	Display a single pole at the origin of the pole-zero map grid.

13	SIN	Display poles at ±i of the pole-zero map grid.
14	ERASE	Erase the transient response curve previously detected by the light pen.
15	INPUT	Begin the input mode.
16	PLACE	Position the cursor.
31	END	Terminate the system.

Appendix D

IBM 2250 PROCEDURES

To Turn Power On

1. Insure that the enable/disable (meter) switch is in the down (disabled) position. This switch is located on the left side of the CRT frame.
2. Depress the "power-on" button located to the right of the CRT. This turns the 2250 power on. The green fuse light (beneath the left side of the alphanumeric keyboard desk) should not be on.
3. Press the machine reset button several times (located inside the left rear panel).
4. Enable the 2250 by setting the enable/disable switch to the up position. The green fuse light should not be off.

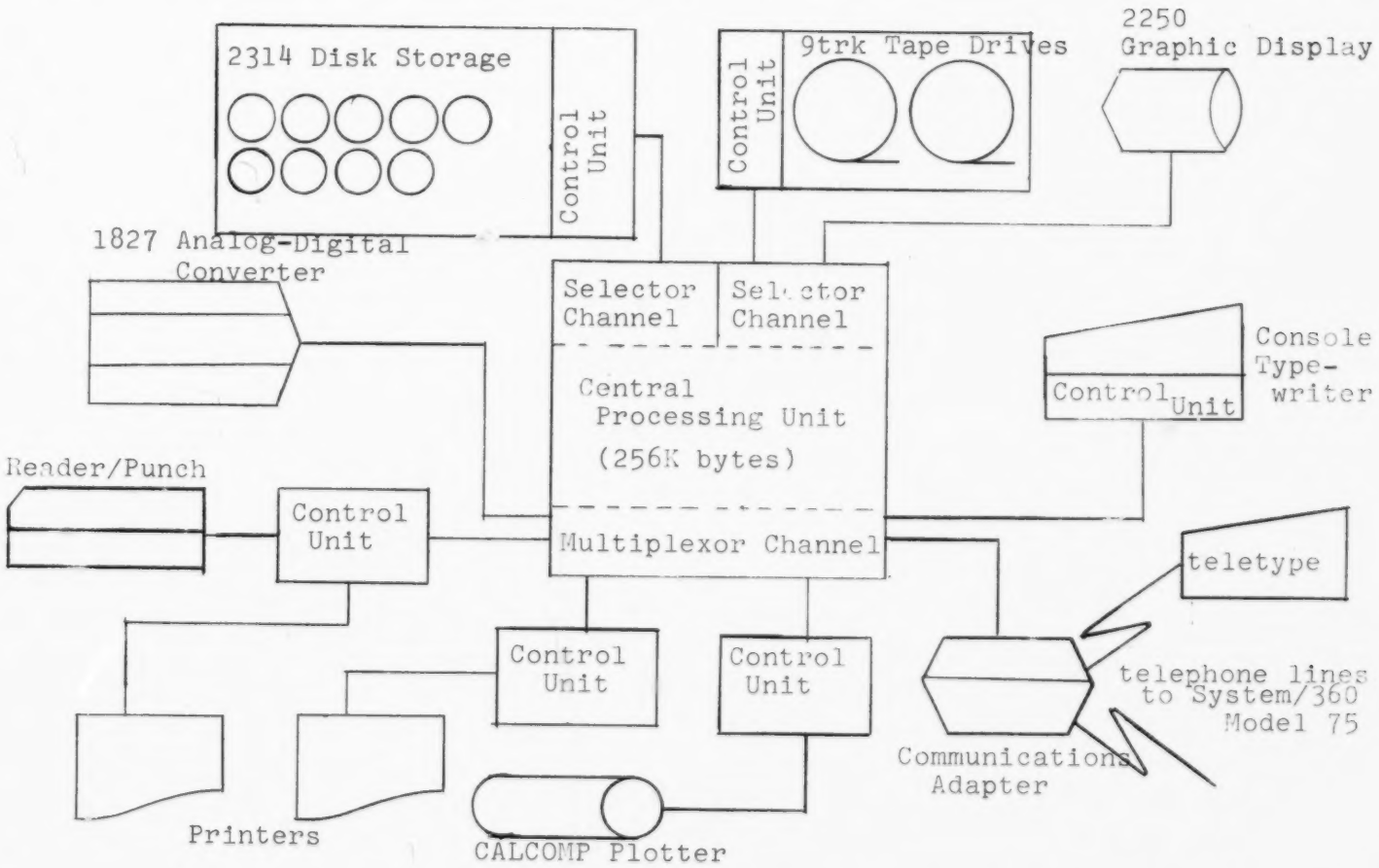
To Release the PZMAP System

1. Hold the intercom switch in the down (talk) position and ask the computer operator to release PZMAP.
2. When the general instructions display appears upon the CRT inform the computer operator via the intercom.

To Turn Power Off

1. Place the enable/disable switch to the down position.
2. Depress the red "power off" button located to the right of the CRT. The machine should turn off in about ten seconds.

Appendix E



Computation Center - University of North Carolina at Chapel Hill
IBM System/360 Hardware Configuration

Appendix F

COMPUTER PROGRAM LISTINGS

(Bound Separately)

BIBLIOGRAPHY

1. Beyer, W. F., "IBM 2250 Text Input Management Routine," TRW Summer Student Assistant Program in Computer Science (Redondo Beach), 1968.

2. Brauer, F., and Nohel, J. A., Ordinary Differential Equations. New York: W. A. Benjamin, Inc., 1967.

3. Cheng, C., Analysis of Linear Systems. Reading, Mass.: Addison-Wesley Publishing Company, 1959.

4. Clark, R. N., Introduction to Automatic Control Systems. New York: John Wiley & Sons, Inc., 1962.

5. Distefano, J. J., Stubberud, A. R., and Williams, I. J., Theory and Problems of Feedback and Control Systems. New York: McGraw-Hill Book Company, 1967.

6. Dorf, R. C., Modern Control Systems. Reading, Mass.: Addison-Wesley Publishing Company, 1967.

7. Friedland, B., and Schwarz, R. J., Linear Systems. New York: McGraw-Hill, 1965.

8. International Business Machines Corporation. IBM System/360 Operating System, Pl/1 Subroutine Library, Computational Subroutines, Form C28-6590-1.

9. International Business Machines Corporation. IBM System/360 Component Description IBM 2250 Display Unit Model 1, Form A27-2270-1, 1967.

10. International Business Machines Corporation. IBM System/360 Operating System Graphics Access Method, Form Y27-7113, 1967.

11. International Business Machines Corporation. IBM System/360 Operating System Graphics Programming Services for IBM 2250 Display Unit, Form C27-6909, 1967.

12. Oliver, A., "A Measurement of the Effectiveness of an Interactive Display System in Teaching Numerical Analysis." Doctoral Dissertation. University of North Carolina at Chapel Hill, 1969.